Envisioning a Future Decision Support System
for Requirements Engineering –
A Holistic and Human-centred Perspective

by

Beatrice Alenljung
ABSTRACT

Complex decision-making is a prominent aspect of requirements engineering (RE) and the need for improved decision support for RE decision-makers has been identified by a number of authors in the research literature. The fundamental viewpoint that permeates this thesis is that RE decision-making can be substantially improved by RE decision support systems (REDSS) based on the actual needs of RE decision-makers as well as the actual generic human decision-making activities that take place in the RE decision processes. Thus, a first step toward better decision support in requirements engineering is to understand complex decision situations of decision-makers. In order to gain a holistic view of the decision situation from a decision-maker’s perspective, a decision situation framework has been created. The framework evolved through an analysis of decision support systems literature and decision-making theories. The decision situation of RE decision-makers has been studied at a systems engineering company and is depicted in this thesis. These situations are described in terms of, for example, RE decision matters, RE decision-making activities, and RE decision processes. Factors that affect RE decision-makers are also identified. Each factor consists of problems and difficulties. Based on the empirical findings, a number of desirable characteristics of a visionary REDSS are suggested. Examples of characteristics are to reduce the cognitive load, to support creativity and idea generation, and to support decision communication. One or more guiding principles are proposed for each characteristic and available techniques are described. The purpose of the principles and techniques is to direct further efforts concerning how to find a solution that can fulfil the characteristic. Our contributions are intended to serve as a road map that can direct the efforts of researchers addressing RE decision-making and RE decision support problems. Our intention is to widen the scope and provide new lines of thought about how decision-making in RE can be supported and improved.

This work has been supported by The University of Skövde and The Swedish Knowledge Foundation.
Acknowledgements

First of all I want to thank my supervisor Prof. Anne Persson for all support and encouragement. You have always given me valuable up-front advice in great things as in small.

I want to express my gratitude to my supervisor Prof. Sture Hägglund and co-supervisor Dr. Pär Carlshamre. You have given me insightful comments on research ideas, been supportive, and asked necessary questions. Furthermore, I want to thank Dr. Tarja Susi who thoroughly reviewed and discussed this thesis. You gave me a lot of helpful comments.

A special thank to my friends and colleagues Tarja Susi, Jessica Lindblom, Åsa Dahlstedt and Lena Aggestam. Thank you for all encouragement and support. You make it fun to go to work.

I also wish to thank my colleagues in the Information Systems Research Group and my colleagues in human-computer interaction and cognitive science.

Last but not least, I want thank my family, my husband Mattias and my sons Sebastian and Zackarias. You are my Everything. Ni är det bästa som finns!
1 Introduction

Researchers and practitioners have for a number of years agreed that requirements engineering (RE) is a critical activity for the success of systems development. The complexity of RE and the large extent of decision-making involved in the RE process have also been recognised. Our fundamental viewpoint throughout this thesis is that RE decision-making can be substantially improved by RE decision support systems (REDSS) based on the actual needs of RE decision-makers as well as the actual generic human decision-making activities that take place in the RE decision processes. Such an REDSS should focus on augmenting the human RE decision-maker’s possibilities to carry out decision-making activities in the whole RE decision process. Currently, there is a lack of understanding of the nature of RE decision-making in practice as well as the problems and difficulties that affect RE decision-makers. In addition, existing RE decision support is limited and mainly concerned with specific RE decision problems on a fairly detailed level. RE decision-making and decision support is a relatively new and immature field of research. Therefore, we provide an holistic and in depth portrayal of RE decision-making as well as suggest desirable characteristics of an RE decision support system. These are intended to serve as a road map that can direct the efforts of researchers addressing RE decision-making and RE decision support problems. Our intention is to widen the scope and provide new lines of thought about how decision-making in RE can be supported and improved.

In this chapter, an overview of the content of the thesis is given together with arguments for the research problem. First, the problem domain is briefly introduced. Second, the research problem along with the research objectives are described and argued for. The research process is then summarised, and the main contributions are listed. Lastly, a thesis outline is presented.

1.1 Problem domain

Requirements engineering (RE) is the “branch of systems engineering concerned with the desired properties and constraints of software-intensive systems, the goals to be achieved in the software’s environment, and assumptions about the environment” (Ebert & Wieringa, 2005, p 453). One of the major challenges of RE is to really understand the needs and characteristics of the domain of which the system will be a part. To be able to develop a system that fulfils the needs of its stakeholders, the purpose of the system needs to be known. If we do not know where to go, then we cannot know how to go there. The “where-to-go-information” includes the needs of the users, the services that the system should provide, and the constraints that need to be taken into account.

Requirements serve as verbalisation of decisions concerning the desired functionality and qualities of a system. Thus, the RE process can be viewed as a decision process and requirements can be perceived as decisions (Aurum & Wohlin, 2003; Evans et
al., 1997). These decisions govern the development process as well as the nature of the product. If inappropriate decisions are made, then both the development process and the system can be negatively affected. The persons who make RE decisions and carry out RE decision-making activities can then be regarded as RE decision-makers.

RE decision-making is complex and of vital importance for the quality of the developed system as well as the systems engineering process. Therefore, RE decision support can increase the effectiveness and efficiency of RE decision-making. The problem domain is elaborated in chapter 2.

1.2 Research problem

Decisions are made throughout the software engineering process, targeting, e.g., requirements, architecture, components, project planning, validation etc. (Kotonya & Sommerville, 1998; Ruhe, 2003a). Similarly, the RE process can be viewed as a decision-making process (Aurum & Wohlin, 2003; Regnell et al., 2001). Decision-making in the RE process is far from straightforward. It is a knowledge-intensive activity, and human decision-makers in general have cognitive limitations (Aurum & Wohlin, 2003). Furthermore, RE decision-makers have to deal with difficulties that stem from the inherent nature of decision-making in natural settings, e.g., uncertain and dynamic environment; shifting, ill-defined, or competing goals or values; time stress; and multiple players (Orasanu & Connolly, 1993). In addition, obstacles such as, e.g., lack of supportive resources, high cognitive load, and pressure need to be managed (Alenljung, 2005). Thus, there are numerous challenges that face RE decision-makers.

Research into the field of RE decision-making and RE decision support is in its infancy (Ngo-The & Ruhe, 2005). A major challenge is to describe and comprehend the nature of RE decision-making, e.g., in terms of which decisions are actually made, which factors affect RE decision-makers, what decision-making activities are carried out, and which decision processes exist. Such understanding makes it possible to effectively improve and support RE decision-making. Thus, more theoretical and empirical research is needed (Ngo-The & Ruhe, 2005). One starting point for such research is general decision-making theories and models of decision processes. This way we can understand the inherent nature of RE decision-making activities (Aurum & Wohlin, 2003; Regnell et al., 2001). Based on knowledge about RE decision-making we can suggest RE decision support and improvements to the RE decision-making process.

Developing support for RE decision-making is a major issue for RE research (Regnell et al., 2001). Ngo-The and Ruhe (2005) argue that RE decision support should not strive for optimality. Many decision situations in RE are so complicated that an absolute optimal solution is not feasible. The decision problems are often “wicked” and contain a substantial amount of uncertainty; meaning that trade-offs based on human judgement are necessary. Instead, the provided support should strive to augment the decision-making capacity of the human decision-maker.
In this thesis we claim that RE decision-making can be substantially improved by RE decision support systems (REDSS) based on the actual needs of RE decision-makers as well as the actual generic human decision-making activities that take place in RE decision processes. For clarification, we use the term REDSS to denote a visionary, non-existing tool and the term RE tool represents existing tools.

Since research into the field of RE decision-making and RE decision support is still immature, a coherent body of knowledge does not yet exist. As far as we know, current research in the field has focused on specific RE decision problems at a fairly detailed level, such as, e.g., requirements prioritisation (Karlsson et al., 1998). This means that most research can be viewed as having a bottom-up perspective. The same tendency can be observed in RE tools, where certain decision support techniques have been integrated. Valuable, although limited, decision support is hence provided. To the best of our knowledge, there is no research into RE decision-making and RE decision support that takes a top-down holistic perspective. Such a perspective can help “draw a map” of the decision situations of RE decision-makers as well as help explore the scope of an RE decision support. Furthermore, to the best of our knowledge, none have conducted empirical in-depth studies in order to give a detailed account of RE decision-making in practice. Our research contributes to filling this void.

Thus, the problems that are addressed in this thesis are the lack of understanding of RE decision-making in practice and the limited RE decision support that is provided today. These problems represent challenges that need to be tackled in order to mitigate them. The aim of our research is to contribute to providing effective RE decision support for RE decision-makers through all of the RE decision processes. In order to take steps towards this aim, the following objectives have guided the research project:

- Give a portrayal of RE decision-making in practice
- Suggest desirable characteristics of an RE decision support system

These objectives pose several questions that need to be answered. The first objective raised, e.g., the following questions. What aspects constitute decision situations? Which RE decision processes exist? What decision-making activities are carried out? Which RE decisions are actually made? What are the characteristics of the RE decisions? Which information sources do RE decision-makers use? Which factors affect RE decision-makers? Which problems and difficulties face the RE decision-makers?

The second objective triggers, for example, the following questions. Which desirable characteristics should an RE decision support system have based on the needs of the RE decision-makers that are derived from examining the practice? Which desirable characteristics should an RE decision support system have based on the nature of the activities in the discovered RE decision processes? Which principles can guide the fulfilment of desirable characteristics? Which techniques, design strategies, design
principles, and approaches can be used in order to realise the suggested characteristics?

The *purpose* is to provide a road map that can direct the efforts of researchers addressing RE decision-making and RE decision support problems. Our contributions provide a widened scope and give new lines of thought about how decision-making in RE can be supported and improved. They can also provide inspiration for RE tool developers that is based on empirical research findings.

We do not claim that it is possible to provide a complete or generalisable description of RE decision-making. Instead, a holistic and in-depth portrayal of RE decision-making in an information-rich case is given. Also, we do not intend to provide solutions concerning specific RE decision matters. Instead, we widen the scope of possible ways of supporting RE decision-makers, by suggesting a range of potential means. Furthermore, we have not taken cost-benefit or other practical aspects into account when suggesting the characteristics. What we have done is taken some steps forward in an immature field, where much research remains to be done. The results of our research can potentially lead into several new research projects.

Our research will, hopefully, in the long term perspective, contribute to helping RE decision-makers experience an improved decision situation. Better RE decision support should also contribute to improving the system engineering process as well as improving the quality of the outcome of that process, the developed system.

The primary *influences* that shaped the focus and content of this thesis are a) the user-centred perspective of the field of human-computer interaction, b) the theoretical foundations of the field of decision support systems, and c) the research problems of requirements engineering. These three fields are integrated in the thesis, although not equally apparent. The principal influences of the research process are constructivism and pragmatism, which required a qualitative research approach as well as a design science approach.

The essence of this argumentation is summarised in Figure 1.
1.3 Research process

The premise of constructivism is that the reality is socially constructed and the world of human perception is not real in an absolute sense (Patton, 2002). The constructivist influence is that we view the experiences of persons and how they perceive their world of essential. To have a pragmatic view obliges us to focus upon consequent phenomena instead of antecedent phenomena, and upon possibilities for actions instead of precedents (Dewey, 1931). This means that the interpretations we make must make sense practically and that we have an interest in “not only for what ‘is’, but also for what ‘might be’” (Goldkuhl, 2004, p 13). Constructivism demands a qualitative research approach in order to explore the world as experienced by RE decision-makers. The quality criteria for qualitative research are not the same as for quantitative research. The appropriate criteria for trustworthiness in qualitative research are credibility, transferability, dependability, and conformability (Lincoln & Guba, 1985). Pragmatism makes a design science approach appropriate, since our purpose is not only to describe decision-making in RE, but also suggest how RE decision-making can be supported.

The research process consists of the following stages: a) literature analysis, b) case study, and c) synthesis. The results of each stage are used in succeeding ones. The research process is extensively described and discussed in chapter 5.

The purpose of the literature analysis was to identify the key aspects which are related to a decision-maker in a decision situation, as well as to obtain the theoretical foundations of human decision-making relevant for decision support. The analysis embraced different types of decision-making theories and literature from decision
support systems. This stage resulted in a decision situation framework that consists of the key aspects related to decision-makers. The framework was used during stage two in order to make sure that all the fundamental aspects are taken into account.

A *case study* was subsequently conducted at a systems engineering company that develops highly advanced systems. This is a contract development context, in which both projects and systems to be developed are large and complex. We have focused on the requirements engineers at a subsystem level. The data collection techniques were open-ended interviews and a focus group session. The interviewees were requirements engineers and their related stakeholders. The decision situation framework was used as a means to structure and guide the data analysis. The results from the case study was a portrayal of the decision situation of RE decision-makers, specifically a) identification of decision matters, decision-making activities, and decision processes in RE, b) information sources used by RE decision-makers, and c) factors that affect the RE decision-maker. It is not possible to paint a generalised picture of RE decision situations, since every organisation is unique and every instance of a decision situation is unique. Generalisation is not a goal of qualitative research. Instead, holistic and in-depth studies of a few information-rich units are obtained.

In the *synthesis*, the empirical findings from the case study were synthesised with work especially from the decision support systems field. This resulted in empirically based desirable high-level characteristics of a visionary REDSS and guiding principles that are empirically as well as theoretically grounded. In addition, available techniques are presented together with the guiding principles as a range of potential means. The purpose of the principles and techniques is to direct further efforts concerning how to find a solution that can fulfil the characteristics.

The research process is the route which has taken us from the objectives that have driven the research project to our main contributions.

### 1.4 Main contributions

At a general level, the combination of three different areas: requirements engineering (RE), decision support systems (DSS), and human-computer interaction (HCI) can be considered to be a contribution in itself (Figure 2). To the best of our knowledge, these fields have not been combined before. The research problems originate from RE. DSS provide “solutions” for those problems. HCI offers the line of thought that permeates the whole research project, i.e. the user-centred perspective and approach. However, we do not use the terminology of user-centeredness, since the term user is often used in RE with a certain meaning. Instead, we frequently use the term decision-maker or human to denote the user we focus on. Hence, we have a decision-maker and human-centred perspective.
At a specific level, there are three main contributions in this thesis: a) a decision situation framework, b) a portrayal of the decision situation of RE decision-makers, and c) desirable characteristics of an REDSS. The most important contributions of these three are the portrayal and the suggested characteristics. The contributions are presented in chapters 6-9. The relations between the contributions are outlined in Figure 3.

The decision situation framework is a theoretically based framework that takes a holistic perspective of the decision situation from the viewpoint of a decision-maker. A decision situation is defined as a contextual whole of related aspects that concerns a decision-maker. The decision situation framework can be used in order to portray a particular decision situation of a certain decision-maker in a human-centred and holistic manner. The framework provided high-level categories which offered a structure to the data analysis process that resulted in the decision situation of RE decision-makers.

The decision situation of RE decision-makers is a portrayal of different aspects related to decision-makers in requirements engineering. Based on an empirical case study, we
describe two different RE decision processes, their decision-making activities and the
decision matters they encounter. The characteristics of the decision matters as well as
the information sources used by RE decision-makers are also elaborated. A number
of factors that directly or indirectly influence RE decision-making and which, as a
consequence, may have an effect on the decision outcome are described. Related to
each factor, difficulties and problems that can cause potential quality problems in RE
decision-making are identified. We studied one case in depth and focused on RE
decision-makers at a subsystem level in a contract development context, where
both projects and systems to be developed are large and complex. Other cases may
identify other details of the decision situation of RE decision-makers, since every
individual situation is unique. Situations can vary along several dimensions, for
example, development context, maturity of the organisation, organisational culture,
or the size of the project. Nevertheless, we claim that our findings consist of the key
aspects of RE decision-making, e.g., decision processes, decision matters, information
sources, and factors. We made several findings within these aspects in the case study.
In future work, new case studies will add findings in terms of new details,
confirming details, nuances, and dimensions along which the RE decision situations
vary. While, future research can be based on the key aspects, much more work is
needed in order to gain a cohesive body of knowledge of RE decision-making. The
decision situation of RE decision-makers is the underpinnings of the desirable
characteristics of RE decision support systems.

Nine empirically grounded desirable characteristics of an RE decision support system
(REDSS) are identified. They are based on the needs of RE decision-makers and the
nature of RE decision-making. This means that we focus on what is generic, and not
on specific RE tasks. For each characteristic, one or more guiding principles that
direct further efforts to find a solution which can fulfil the characteristic are
suggested. The guiding principles are empirically and theoretically grounded. For all
guiding principles we also present some available techniques, which are theoretically
grounded. In order to illustrate a way to use the characteristics in the future, we have
taken a step further and developed an evaluation method called DESCRY. The
purpose of method is to investigate to what extent RE tools have decision-supporting
capabilities.

The contributions resulted in the following publications:

making in the software engineering process: A position paper. In: B. Regnell, E.
Kamsties & V. Gervasi (Eds.) Proceedings of the 10th International Workshop on
Requirements Engineering: Foundation for Software Quality, REFSQ 2004 (pp 63-68),
7-8 June 2004, Riga, Latvia

perspective: A framework for analysing decision situations. In: P. Backlund, S.
Carlsson & E. Söderström (Eds.) Proceedings of the 4th International Conference on
Business Informatics Research, BIR 2005 (pp 13-22), 3-4 October 2005, Skövde,
Sweden


There are also some manuscripts submitted:

Alenljung, B. & Persson, A. (200x) Portraying the practice of decision-making in requirements engineering: A human-centred perspective (submitted for journal publication)

Alenljung, B. & Persson, A. (200x) Beyond fragmented tools: Towards a holistic and human-centred decision support system for requirements engineering processes (submitted for journal publication)


1.5 Thesis outline

Chapters 2, 3 and 4 contain the theoretical foundations. Chapter 5 provides an account of the research process. Chapters 6, 7, 8, and 9 report the results, and chapter 10 concludes the thesis with a discussion of its content. The remainder of this thesis is structured as follows:

Chapter 2 describes the field, requirements engineering (RE), which is the domain that we primarily contribute to. A brief introduction to RE is given and the notion of requirements is presented. The activities of RE as well as RE tools are discussed, and RE as decision-making is elaborated.

Chapter 3 contains theories of human decision-making, which were used when the decision situation framework was created and used as a basis for the case study. The concept of decision is discussed. Different classes of decision-makers are presented along with psychological types and decision styles of decision-makers. Decision-making as a human activity, which includes theories of problem solving; models of decision processes; theories of individual, behavioural decision-making; theories of group decision-making; and theories of organisational decision-making is described.
Chapter 4 discusses the concept of decision support systems and their characteristics. Different types of decision support systems are discussed as well as the various ways decision-making can be supported. Benefits and limitations of decision support systems are also mentioned. Input from the field of decision support systems was used in the development of desirable characteristics and guiding principles for REDSS.

Chapter 5 reports the research process and the methods used. This includes a discussion of the methodological considerations. In addition, the research process is presented, which includes a literature analysis, a case study, and a synthesis. Reflections on the case study, in particular, are given.

Chapter 6 presents the generic decision situation framework, which guided the analysis of the empirical data that is the ground for the decision situation of RE decision-makers.

Chapter 7 portrays the decision situation of RE decision-makers, including RE decision processes, RE decision-making activities, RE decision matters and their characteristics, and the information sources used by RE decision-makers. These parts of the decision situation were used in the development of the desirable characteristics of an REDSS derived from the nature of RE decision-making activities.

Chapter 8 continues the depiction of the decision situation of RE decision-makers, by showing factors that affect the RE decision-maker. The factors are attitudes toward requirements work, communication and coordination, resource, pressure, cognitive load, and knowledge. These parts of the decision situation were used in the development of the desirable characteristics of an REDSS derived from the needs of RE decision-makers.

Chapter 9 elaborates the desirable characteristics of an REDSS. The characteristics are reduce the cognitive load, ensure high usability, support availability of different types of information, support creativity and idea generation, support knowledge sharing and transfer, support idea evaluation and problem solving, support decision communication, and support coordination. The characteristics and their guiding principles are the foundation of the evaluation method.

Chapter 10 provides a discussion of the research results. The contributions are reviewed, and reflections on the results are given. The thesis concludes with an outline of future work.
2 Requirements engineering

This chapter presents an overview of requirements engineering (RE). A brief introduction of the field is provided, and the notion of requirements is explained. A presentation of the RE process and its general activities is followed by a discussion of RE tools. In addition, RE as decision-making is elaborated, and finally, the chapter is summarised.

2.1 Introduction to requirements engineering

The development of interactive systems, such as ticket selling machines or business intelligence systems, as well as embedded software, e.g., real-time software, is an expensive and complex process. Therefore, it is important that the process is successful in terms of each system component fulfilling its purpose and requirements. However, to define requirements is by no means a trivial task in the systems engineering process. In fact, it is one of the most critical and error-prone parts of systems engineering. One of the major challenges of requirements engineering (RE) is to understand the needs and characteristic of the domain in which the system is going to be a part. The potential domains are as shifting as the world around us. It is therefore impossible for a developer to be knowledgeable about everything from health care to the military domain, or from pleasure and entertainment to work settings.

To examine the domain of interest and define what the system should do and what qualities it should have is the role of RE in systems engineering. “Requirements engineering (RE) is the branch of systems engineering concerned with the desired properties and constraints of software-intensive systems, the goals to be achieved in the software’s environment, and assumptions about the environment” (Ebert & Wieringa, 2005, p 453). Thus, the RE process is an intrinsic part of the software engineering process, which in turn is a natural part of systems engineering (see Figure 4). Software engineering is an “engineering discipline that is concerned with all aspects of software production from the early stages of specification to maintaining the system after it has gone into use” (Sommerville, 2007, p 7). Systems engineering is the “activity of specifying, designing, implementing, validating, deploying and maintaining socio-technical systems. Systems engineers are not just concerned with software but also with hardware and the system’s interactions with users and its environment” (Sommerville, 2007, p 25).
Figure 4, RE is an intrinsic part of the process of software-intensive system or product development

These engineering processes are active during the whole lifecycle of the system or product. In this thesis, we use both the term system and the term product. These terms always denote software-intensive systems or products. We are aware of the fact that a systems engineering process and a product development process are not necessarily comparable in all aspects. However, in this thesis those differences and similarities are not essential, and thus we do not elaborate this. We use system and product almost as synonyms. The focus of concern is RE, and RE processes as such vary due to several factors (see section 2.3.1).

Three important goals of the RE process (Pohl, 1994) are a) to develop a complete system specification starting from vague ideas about the system to be, b) to change informal knowledge about the system into formal representations, and c) to negotiate agreement on the specification from a number of personal views.

Decisions are made at all steps in systems engineering, for example, with regard to requirements, architecture, components, project planning, validation etc. (Kotonya & Sommerville, 1998; Ruhe, 2003a). In the same way, the RE process can be viewed as a decision-making process (Aurum & Wohlin, 2003; Regnell et al., 2001). Decision-making in the RE process is far from straightforward. It involves the difficulties that characterise decision-making in natural settings, e.g., uncertain and dynamic environment; shifting, ill-defined, or competing goals or values; time stress; and multiple players (Orasanu & Connolly, 1993). This implies that decision-makers in the RE process need decision support, for example, RE tools adapted to the actual decision-making activities.

In order to be able to purposefully improve RE decision support it is essential to gain a comprehensive understanding of the RE decision-makers’ decision situation, e.g., in terms of which decisions are actually made, which factors affect RE decision-makers, what decision-making activities are carried out, and which decision processes exist? This understanding enables a definition of what kind of decision support RE decision-makers need and what should constitute such support? For
clarification, in this thesis we use the term RE decision support system to denote a visionary, non-existing tool and the term RE tool represents existing tools.

The first step towards a human-centred RE decision support system is to walk through the state-of-the-art of the RE field. First of all, we elaborate on the concept of requirement.

2.2 Requirements

An essential concept in RE is that of requirement. A requirement states what is demanded of a system or product. In this section, the notion of requirements is defined, the different types of requirements are elaborated, and the concept of requirements specification is presented.

2.2.1 Definition

There are several definitions of the concept requirement. Sutcliffe’s (2002, p 1) definition is that “requirements involve finding out what people want from a computer system, and understanding what their needs mean in terms of design”. This definition summarises, in general terms, what it is all about. It shows that requirements convey the purpose of the system and thereby the requirements direct the development of the system. Other definitions explain what requirements include. For example, Sommerville (1995, p 64) defines a requirement as “a statement, in natural language plus diagrams, of what services the system is expected to provide and the constraints under which it must operate. It is generated using customer-supplied information.” Another example is Kotonya and Sommerville’s (1998, p 4) definition that “requirements define the services that the system should provide and they set out the system’s operation”. A more extensive definition is the IEEE 610 standard which views a requirement as (Machado et al., 2005, p 47):

1. “A condition or capability needed by a user to solve a problem or achieve an objective;
2. A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents;
3. A documented representation of a condition or capability as in (1) or (2).”

As exemplified by the examples above, there is no agreed definition of the notion of requirement. However, the essence of the definitions presented is the same. They show that in order to be able to develop a system that fulfils the needs of its stakeholders, we have to know the purpose of it. If we do not know where to go, then we cannot know how to go there. The “where-to-go-information” includes the needs of the users, the services that the system should provide, and constraints that need to be taken into account.

A requirement can be regarded as a representation of a specific decision concerning a particular aspect of a system. These decisions govern the development process as well as the nature of the product. If an inappropriate decision is made, then both the
development process and the product can be negatively affected. There is a wide variety of decisions communicated via requirements, since requirements are multifaceted and can be of different types.

2.2.2 Types of requirements

There are a number of ways to categorise requirements. A common way to categorise requirements is:

- Functional requirements
- Non-functional requirements

Functional requirements state the system's functionality, i.e. what the system will do. This includes the behaviour of the system, what it contains and its components. Non-functional requirements are concerned with the qualities of the system, which means they limit the possible solutions through, for instance, security or performance requirements (Aurum & Wohlin, 2005; Sutcliffe, 2002; Kotonya & Sommerville, 1998).

Non-functional requirements can be sorted into three groups (Sommerville, 2004):

- Organisational requirements
- Product requirements
- External requirements

Organisational requirements originate from the developer’s or a customer’s organisation and they set the limits of the process of developing the system, for example, with regard to delivery, implementation, and standard requirements. Product requirements specify the characteristics of the desired system, such as usability, reliability, portability, and efficiency requirements. External requirements are derived from the environment of the system and development process. They are categorised as legislative, ethical, or interoperability constraints (Sommerville, 2004). There is no clear distinction between the categories and it is often possible to put multiple labels on a requirement. Despite this, especially functional and non-functional requirements are often used for categorisation (Aurum & Wohlin, 2005; Kotonya & Sommerville, 1998).

In a categorisation made by Sutcliffe (2002), functional as well as non-functional requirements can be characterised as being goals, which is a general term for a formulation that states what to achieve. The other categories are attributes and constraints. Attributes are the system’s qualities or properties. Constraints are aspects the system has to take into consideration, since they delimit the system. There are, for instance, physical, environmental, cost, or legal constraints.

Another way of categorising requirements is (Aurum & Wohlin, 2005):

- Goal level requirements
- Domain level requirements
- Product level requirements
- Design level requirements
Goal level requirements are concerned with business goals. Domain level requirements relate to the problem area. These types can be compared to general requirements, which are described by Kotonya and Sommerville (1998), and system requirements, presented by Sutcliffe (2002) as requirements that in broad terms declare what the system should do. Such requirements can, for example, emerge from organisational needs. The boundaries of the different kinds of requirements are not clear. The product level requirements are linked to the product. Sutcliffe uses the term software requirements for this type, which includes software functions. The design level requirements consist of statements of what should be constructed. The design level requirements are similar to, what Kotonya and Sommerville calls, implementation requirements that specify how the system should be implemented. Then, there are a number of other ways to group requirements.

What is the point of requirements categorisation? For example, during establishment and analysis of requirements the different categories can be useful as guidance and to check that all necessary types of requirements are covered. What the necessary types of requirements are may vary. It can, for instance, be necessary for an organisation or a project to decide upon which types of requirements that need to be covered. This can reduce the amount of missing requirements which imply missing decisions. If decisions concerning a certain type of requirements are missing, e.g., usability, then the outcome of the systems engineering process risks being of low quality.

Requirements decisions are important and they should be made by those most competent in this matter and authorised to make them. Otherwise, other system development workers, e.g., designers, programmers and testers, risk making ad hoc decisions concerning those aspects.

Irrespective of which type of requirement it is, it has to be specified in a clear way so that its intended readers obtain useful information to act upon in their respective roles.

2.2.3 Requirements specification

Requirements specification is a term given two different meanings in the literature. It is used as either describing the RE activity of specifying requirements which can be understood by their stakeholders or as the name of the document covering the requirements, i.e. a complete description of what the system should do (Matulevičius, 2004). From our point of view, this is not a contradiction, but rather “two sides of the same coin”. Someone has to carry out the activity in order to receive a collection of specified requirements. To divide the term into two different ones will not add any value to this thesis. Thus, we include both meanings in our use of the term, and let the textual context show the current meaning.

It is important, to not only specify requirements of different types, but to write clear, understandable, and unambiguous requirements. Otherwise, the users of the requirements may interpret them in an incorrect way and thereby use them inappropriately. Requirements can be written in natural or formal language.
Natt och Dag and Gervasi (2005) advocate natural language when requirements are specified, which they do for communicative reasons. Natural language is the primary communication language of people, which makes it more understandable for more readers than formal language. Natural language is useful for validation of requirements, and is not domain-specific or specific for a certain level of abstraction. Therefore, it is more flexible than formal language (Natt och Dag & Gervasi, 2005). On the other hand, natural language can be, and often is, ambiguous and difficult to understand. This can lead to misinterpretations (Sutcliffe, 2002; Kotonya & Sommerville, 1998), and as a result problems occur during the system development process causing quality problems in the system.

It is necessary to specify requirements in order to develop successful systems, but low quality requirements reduce the chances of reaching that goal. Difficulty understanding the requirements is one such quality problem. However, this is not the only threat to quality. Other common requirements problems, listed by Kotonya and Sommerville (1998), are, e.g., inconsistent and incomplete requirements as well as wrong requirements, e.g., they do not fulfil the needs of the users. Other essential problems are that it is expensive to make changes among agreed requirements, and that there are confusions and mix-ups between different requirements stakeholders (Kotonya & Sommerville, 1998).

A reflection of specifying requirements at different levels is the importance of traceability and dependencies between the requirements at different levels and within each level (see e.g. Dahlstedt & Persson, 2005). To make decisions concerning, for instance, a requirement change proposal at an organisational level will most likely have an effect on other requirements. There are also cascade effects concerning the traces between the levels. A requirement at an organisational level is most likely concretised in several requirements at the project level. This can also be the reverse case, since a requirement at a project level can be a concretisation of several requirements at a higher level. The decision-maker needs to have information about these relationships in order to make a well-informed decision, where at least some consequences are known and can be taken into account.

However, trade-offs may be needed between how much information is necessary and the increased complexity and amount of administrative resources needed to keep the information up to date. When the relationships and dependencies become complex, it might be necessary to visualise them, so that the RE decision-maker more easily can see the consequences of, for instance, a requirements change.

Accordingly, since requirements can be regarded as representations of essential decisions concerning the system as well as the system engineering process, the activities carried out in the RE process can be viewed as decision-making activities. If the quality of decision-making improves, then the quality of the RE process will also improve.
2.3 Activities in requirements engineering

The process of requirements engineering (RE) consists of different, but interrelated, interdependent, and iterative, activities. In this section, an overview of the RE process is provided, and the different activities in the process are also described.

2.3.1 The requirements engineering process

An RE process has several activities through which different kinds of information flow and knowledge increases, not least concerning requirements (see Figure 5). On a general level the RE process consists of the activities: elicitation, analysis, negotiation, validation, documentation, and management. In requirements elicitation, the requirements are discovered. These “raw” requirements need to be refined, which is conducted during requirements analysis. There are multiple stakeholders involved in the process who have different views and needs. Thus, requirements negotiations are necessary in order to agree on a set of requirements. The requirements should also be validated to ensure the quality of the future system. The requirements are documented, and requirements management is also necessary.

There are of course different viewpoints concerning exactly which activities should be listed and how they should be termed (see e.g. Bray, 2002; Eriksson, 2007; Kotonya & Sommerville, 1998; Sawyer, 2005). The ones we have chosen in this thesis are in some way included in all RE process descriptions we have seen in the literature.

The RE process is highly iterative, although all activities are not necessarily “active” in all instances of iterations. In the “wheel” of RE, the activities are intertwined and mutually dependent, which is discussed further on in the chapter.

![Figure 5, The iterative RE process with its interrelated and interdependent activities](image-url)
RE is a natural part of the systems engineering process. However, the process takes place within a wide variety of developmental and organisational contexts. The way RE activities are carried out needs to be adjusted to the characteristics of the context at hand (Sutcliffe, 2002). There are several factors that affect the RE process, for instance, the experience of the requirements engineers and the technical maturity of the organisation. Other factors are the organisational culture, the type of engineering and managerial disciplines prevailing, or the maturity of the RE process (Kotonya & Sommerville, 1998; Sawyer, 2005).

The RE process also depends on the development context and to what extent there is adaptability to users or future use (see Figure 6). There are different kinds of development contexts, where the market dimension is oriented from a narrow targeted product i.e. in-house and contract development, to wider ranging products i.e. market-driven development. In-house and contract development is customer-specific, in contrast to market-driven which is not. Adaptability range from user-specific development, e.g., bespoke systems, to “future use” development, e.g., commercial-of-the-shelf (COTS) products (Sutcliffe, 2002). By future use development we mean that the development process is driven by some form of prospective use. The product is indented to be part of different kinds of situations of use, in contrast to the user-specific development that has more defined and specific situations of use to take into account.

In *in-house* development, the product or system is adapted to the needs and characteristics of a specific organisation. Such systems are called a *bespoke* system. The organisation is known by the requirements engineers, who have the users and the domain available. The RE work can be initiated by, for example, a policy route or problem in an existing system (Sutcliffe, 2002).

*Contract* applications are developed for a specific customer and they are often large and expensive systems. The systems engineering process often starts with a request for tender by the customer. RE that concerns contract applications is performed from two perspectives. RE can be necessary both for the customer organisation and the development organisation. Contract applications are similar to in-house systems in
that they have particular organisations and particular users to address the system to. On the other hand, RE in contract applications entails more negotiations, trade-off analyses and prioritisation (Sutcliffe, 2002).

*Market-driven* RE has other characteristics than customer-specific RE, which are summarised by Regnell and Brinkkemper (2005). The developing company is the only risk-taker in such a project. The product is often delivered in several releases, which causes extra attention to be placed on release planning, particularly on time-to-market and return-on-investment. The set of requirements and the product are continuously extended and evolved. The requirements are often stated informally and are active in the whole product lifecycle (Regnell & Brinkkemper, 2005).

A contract application can be viewed as a bespoke system, in the sense that it is tailored to the customer. Sutcliffe (2002) mentions that research within the RE community has traditionally been focused on bespoke systems. However, bespoke systems receive lesser attention in the RE research community, since legacy systems, system evolution and reuse become more and more important (Sutcliffe, 2002). Both in-house and contract development can involve legacy systems, system evolution, and reuse. They can also include COTS.

Instead of developing a new system, the use of *COTS* products can be an alternative. A *COTS product* is a “commercially available or open source piece of software that a software project can reuse and integrate into their own products” (Torchiani & Morisio, 2004, p 91). RE is conducted from two perspectives concerning COTS products. The developer has initially carried out RE activities in order to create a product that is attractive to the market. The customer conducts RE activities when selecting a suitable COTS product. Compared to in-house, bespoke systems, the developers of COTS products often do not have a target organisation available and there are fewer stakeholders from which to elicit requirements. Instead, the requirements engineers must be creative and often a system vision is the aim (Ebert & Wieringa, 2005; Sutcliffe, 2002). According to Ebert and Wieringa (2005), a *COTS-based system* is one that consists of COTS products. The development of a COTS-based system can be either in-house, contract or market-driven development. The development of COTS products is market-driven.

If we claim that the RE process is a decision-making process, then we can argue that the RE process can also be improved in terms of better decision-making. This means that the potential of positive consequences of implemented decisions increases. Positive consequences can be, for example, increased user satisfaction, reduced implementation costs, or products that are better adapted to the market. It also means that the decision-making activities within the process are conducted in a cost-efficient way. To improve decision-making, we also need to identify and understand the obstacles of RE decision-making. Otherwise, we will not know what challenges to address or how to support RE decision-making.

The variability of RE processes causes differences in how the activities are carried out. Which aspects of an RE activity that are given extra attention depends on the
type of RE process. This implies that the need for RE support varies depending on
the context. Different types of information may need to be expressed, analysed and
presented. The need for RE decision support may also vary in the same way. This
does not necessarily mean that a certain RE tool should be dedicated to a particular
variant of RE. However, flexibility in how to use the tool is, most likely, of vital
importance.

There are a number of aspects that need to be dealt with during all RE activities;
requirements elicitation, requirements analysis, requirements negotiation,
requirements validation, requirements documentation, as well as requirements
management. The first activity described is requirements elicitation.

2.3.2 Requirements elicitation

A central activity in requirements engineering (RE) is to generate the requirements of
the system. In requirements elicitation the needs of the users and other stakeholders
should be learned and understood in order to communicate this between developers.
The sources of requirements are of different types, such as stakeholders’ opinions,
documentation, and existing system (Kotonya & Sommerville, 1998; Zowghi &
Coulin, 2005). Requirements elicitation can be compared to data collection, i.e. data
concerning relevant aspects are brought together. “Data collection” has to be
conducted several times within the RE process, since it is not possible to “collect” a
complete set of relevant “data” at one time. The world is dynamic and so is the
relevant set of “data”. The comprehension of what “data” is needed during an RE
process also emerges while working with the process. To put it in the words of
Zowghi and Coulin (2005, p 21) requirements elicitation “must allow for
communication, prioritization, negotiation, and collaboration with all the relevant
stakeholders. It must also provide strong foundations for the emergence, discovery,
and invention of requirements as part of a highly interactive elicitation process.”

There are five fundamental types of activities in the process of requirements
elicitation (Zowghi & Coulin, 2005):

- Understanding the application domain
- Identifying sources of requirements
- Analysing the stakeholders
- Selecting the techniques, approaches, and tools to use
- Eliciting the requirements from stakeholders and other sources

*Understanding the application domain* concerns investigations of the setting in which
the system is going to be a part. For example, if a system is meant to support the
work processes in an organisation, then we need information about, e.g., key
business goals and issues, work processes, as well as organisational, political and
social aspects in relation to the system (Zowghi & Coulin, 2005). For example, if a
computer game is to be developed, then there can be other types of information that
are needed, such as domestic settings, or game genre characteristics.
The necessary information can seldom be found using just one source, and to complicate matters it is not always clear which sources are relevant. Instead, the sources of requirements have to be identified. Examples of sources are stakeholders, existing systems and processes, and existing documentation (Zowghi & Coulin, 2005). The specific types of sources can differ depending on what kind of system or product is under development. If a product, based on more non-traditional concepts, e.g., ubiquitous computing or tangible computing, is built, then there are perhaps no existing systems available to use as a source. If a new version of a management information system is going to be implemented in a particular organisation, then the existing system is most likely a valuable source.

A system always has stakeholders, i.e. the categories of persons that are affected by it or have an interest in it. The characteristics and the needs of the stakeholders differ depending on what is to be developed. Thus, the stakeholders should be analysed. Examples of stakeholders are project sponsors, product partners, and users. It is important to analyse and involve the stakeholders (Zowghi & Coulin, 2005). The involvement of stakeholders is not always easy, specifically when it comes to users. It is probably easier to involve users when a new version of a system in a particular organisation is to be implemented, compared to involving, e.g., handicapped children when developing a software-intensive pedagogical aid. Different analysis and involvement approaches are necessary.

Hence, appropriate techniques, approaches, and tools to use have to be selected. There is no panacea for the elicitation process. Instead, careful selection is necessary. Often a combination of several techniques, approaches, and tools are more effective than just single choices (Zowghi & Coulin, 2005).

The elicitation of requirements from stakeholders and other sources is made when we know what the sources are and how to perform the elicitation of core requirements (Zowghi & Coulin, 2005). This must be conducted in an ethical way. Since there are always people involved, ethical considerations, such as confidentiality and respectfulness, are essential.

The variety of different instances of requirements elicitation processes calls for a large and flexible tool box, i.e. there is a need for a wide range of techniques and approaches in requirements elicitation. There are general data collection techniques, such as interviews, questionnaires, task analysis, domain analysis, introspections, brainstorming, observations, think aloud protocols, and documentation analysis (Kotonya & Sommerville, 1998; Sutcliffe, 2002; Zowghi & Coulin, 2005). More specific techniques have also been proposed, such as, e.g., card sorting, laddering, joint application development, apprenticing, prototyping, and scenarios (Kotonya & Sommerville, 1998; Zowghi & Coulin, 2005). Furthermore, there are approaches that can be used in the requirements elicitation phase. Zowghi and Coulin (2005) mention, for instance, requirements workshops, goal-based approaches, and viewpoints. Kotonya and Sommerville (1998) also discuss requirements reuse as an approach for eliciting requirements.
From a decision-making process perspective, requirements elicitation is important for several reasons. A) Since, we can regard requirements as expressions of decisions, decision alternatives are generated in this activity. Existing “solutions”, i.e. reuse of existing requirements, can also be identified. Thus, the quality of decision alternatives is dependent on the quality of requirements elicitation. B) Relevant information is gathered in this activity; information that constitutes an important base, e.g., for evaluation of decision alternatives. Without proper information, or if the information is of low quality, then this will most likely have a negative effect on the choice of decision alternative to implement. C) RE activities in general and requirements elicitation in particular, can be viewed as learning processes that increase the body of knowledge regarding important aspects. Proper knowledge improves the possibility of RE decision-makers performing well-grounded judgements and arguments concerning decision alternatives. Perhaps it would be a fruitful way to combine human learning theories with RE activities and thereby find new techniques or guidelines for how to, for example, carry out requirements elicitation. Just like requirements elicitation can be viewed as a learning process, it can also be considered to be a knowledge management process (see e.g. Jennex, 2005). Perhaps the knowledge management domain can be a successful “partner” to current RE practices.

Requirements elicitation is not an activity that is conducted “once and for all”. Instead, it can be carried our several times in the life cycle of a product. Moreover, it is interrelated with requirements analysis, since, as Sutcliffe (2002) puts it, when requirements are elicited some analysis is inevitably carried out.

### 2.3.3 Requirements analysis

While, requirements analysis and requirements elicitation are interdependent and iterative (see Figure 7), none of these activities have high value on their own. In the previous section we compared requirements elicitation with data collection. The collected data has to be analysed in order to be understood and interpreted. Accordingly, requirements analysis can be compared to data analysis. We have to organise, scrutinise, and add meaning to the “data”, in order for it to be useful. During “data analysis” we become aware of missing information and insufficient knowledge. Hence, the “data analysis” drives further “data collection” efforts. Requirements elicitation and requirements analysis can be conducted almost synchronously or at separate times.

![Figure 7, The close and iterative relation of requirements elicitation and requirements analysis](image-url)
If the output of requirement elicitation is viewed as raw requirements and raw related information, the output of requirements analysis is refined requirements and refined related information.

According to Kotonya and Sommerville (1998), the purpose of requirements analysis is to establish a complete and consistent set of requirements. The need for a requirement has to be established and it must be ensured feasible within the budget and schedule. The draft requirements document should be scrutinised in order to find missing requirements, requirements conflicts, ambiguous requirements, as well as overlapping requirements (Kotonya & Sommerville, 1998). However, not only requirements should be the result of requirements analysis. Sutcliffe (2002) claims we need information in order to understand the system context and to be able to model and write scenarios. There are different types of information that should be gained during this activity: a) dynamic information, b) static information, c) contextual information, and d) intentions. While dynamic information describes events, actions, procedures, and tasks, static information is, for example, entities, agents, attributes, relationships, properties and states. Contextual information concerns the setting of the system, while information about intentions includes goals, arguments and justifications. This information can be used, e.g., for refinement of requirements, interpretation, modelling, and design (Sutcliffe, 2002).

There are several techniques for conducting requirements analysis. Analysis checklists consist of questions that can be used to systematically walk through the requirements. Another technique concerns interaction matrices, which present interactions between requirements and support the discovery of conflicts and overlaps (Kotonya & Sommerville, 1998). Sutcliffe (2002) presents some analytic techniques, e.g., card sorting, repertory grids, and laddering. He also describes two general approaches for analysis; a) top-down decomposition that focuses on goal analysis and b) bottom-up analysis that concerns event analysis.

Similar to requirements elicitation, the quality of requirements analysis affects the quality of RE decision-making. The most important success factor for requirements analysis is probably the experience and knowledge of those who carry out the analysis.

Another reflection is that it would be interesting to investigate to what extent different types of information need to be stored together with and related to specific requirements in order to support RE decision-making. It would also be interesting to see if current RE tools afford storing of different kind of information and if the relations between information and requirements are represented in an efficient and effective manner, so that the RE decision-makers can interpret the relations correctly and easily.

As previously argued, it is not possible to state requirements once and for all. One of several reasons is the challenge of dealing with different stakeholders’ opinions and needs, which can change during the systems engineering process. There are always
multiple stakeholders involved, which necessitate requirements negotiation activities.

### 2.3.4 Requirements negotiation

The stakeholders of a system and a system engineering process have different perspectives, goals, and needs. This inevitably has an effect on the requirements, and trade-offs between these conflicts are necessary (Grünbacher & Seyff, 2005; Kotonya & Sommerville, 1998; Sutcliffe, 2002). This is handled in the activity called requirements negotiation, which is a process where requirements conflicts are discussed and solved through compromises (Kotonya & Sommerville, 1998). Requirements negotiation can iteratively occur several times in the RE process. It does not take place in a specific order in relation to other RE activities. Negotiations can also be made during other activities, for example, if a goal analysis is conducted cooperatively between stakeholders, then negotiations concerning the relative importance of goals are certainly needed.

An important purpose of requirements negotiation is to agree on a set of requirements. More specifically, as described by Sutcliffe (2002), there should be agreement with regard to the most appropriate design options and also with regard to trade-offs between conflicting requirements. There should also be a selection of requirements for prioritisation. Grünbacher and Seyff (2005) mention that another important result of requirements negotiation is an understanding of why there is a disagreement among the stakeholders. Such disagreements are threats and they need attention and should be dealt with in the project management.

Grünbacher and Seyff (2005) present additional benefits of the requirements negotiation process. The process yields an awareness of the project constraints among the stakeholders, which is critical for the project in order for it to be successful. Another benefit is that negotiation can facilitate the management of requirements changes, since the stakeholders become acquainted with current issues and alternatives. Negotiation can also enhance team learning, since stakeholders have different backgrounds and roles. In discussions concerning aspects of the requirements, the stakeholders can learn from each others’ perspectives. For example, the developer can learn about the user’s domain and the user can learn about technical possibilities and constraints. The negotiation process also makes it possible to reveal tacit knowledge. Hidden issues and assumptions are raised in the requirements discussions. Another benefit of negotiations is that certain techniques support management of requirements complexity. A related problem is uncertainty, which can be reduced by drawing attention to important aspects and creating a shared perspective among the stakeholders. Finally, better solutions can be reached, because negotiation forces stakeholders to understand the whole view. This overview reduces the risk of suboptimal solutions (Grünbacher & Seyff, 2005).

Sutcliffe (2002) further discusses that context and domain information is needed in order to deal with the core of negotiation, i.e. discussion, explanation, and management of requirements conflicts. Such information is vital in order to
understand the reason for conflicts. For instance, stakeholders’ viewpoints can be represented in models and lists, while a comparison of viewpoints can be made easier with tables or matrices. Two examples of particular techniques are the House of Quality (Hauser & Clausing, 1988 in Sutcliffe, 2002) and the analytic hierarchy process (Ryan & Sutcliffe, 1998 in Sutcliffe, 2002).

Similar to all other parts of the RE process, requirements negotiation has its difficulties which must be dealt with. According to Kotonya and Sommerville (1998), the negotiation process should be an objective process, but it seldom is. The decisions should be based on technical and organisational needs, and logical arguments should be put forward. However, political and personal aspects influence the outcome of the negotiation (Kotonya & Sommerville, 1998).

Stakeholders usually have different opinions and make different judgements concerning decision alternatives. Thus, decision negotiations in RE decision-making processes are inevitable. As a consequence, RE decision support should not only facilitate decision-making involving one person, but also aid group decision-making activities.

In practice, requirements negotiation is often closely linked to other parts of the RE process, such as requirements validation (Sutcliffe, 2002).

2.3.5 Requirements validation

The term validation has slightly different meanings for different authors in the literature. Sutcliffe (2002, p 54) describes that validation “involves getting users to understand the implications of a requirements specification and then agree that it accurately reflects their wishes”. This statement can be contrasted to the view of Kotonya and Sommerville (1998, p 87) who define requirements validation as being “concerned with checking the requirements document for consistency, completeness, and accuracy”. Kotonya and Sommerville (1998, p 87) further write that “the aim of requirements validation is to ‘validate’ the requirements, i.e. check the requirements to certify that they represent an acceptable description of the system which is to be implemented”. Sutcliffe is more oriented towards the users and how to get the users to comprehend the consequences of the requirements. We consider the user perspective important, although not enough. There are other stakeholders important in requirements validation, for example, a customer or another project sponsor. An obstacle of user requirements validation is that there are not always real users available. The availability depends on what type of application that is developed. The availability of users is most likely higher in the development of in-house bespoke systems, than market-driven development.

Bray (2002) as well as Kotonya and Sommerville (1998), on the other hand, focus more on the requirements specification per se and detecting errors. Thereby, the priority is to obtain high quality documents. It is important to “ensure that problem domain characteristics and system requirements are fully and precisely determined; and that the specification unambiguously defines a system behaviour such that,
given the problem domain characteristics, it will satisfactorily meet the various requirements” (Bray, 2002, p 210). Thus, there are differences in what gets attention. However, both perspectives are important and therefore we do embrace both in our view of validation.

In Sutcliffe’s (2002) view of validation, it is often conducted in workshops with users and designers. In such a workshop, the future system is walked through and critiqued. The system can be represented, e.g., by models expressed with a notation or a prototype. The facilitator can, for example, use scenarios and with their help show the behaviour of the system and its inputs and outputs (Sutcliffe, 2002). An important challenge is to get the users to really understand the consequences of the decisions that are represented by the requirements. The users’ comprehension processes can be facilitated by concretisation. This can be considered a paradox. RE begins with concrete data which is transformed into abstractions. Then when it comes to validation, we do the opposite and go from the abstract to the concrete. Abstract representations can be difficult to understand sufficiently (Sutcliffe, 2002). Prototypes are a good way of helping users and other stakeholders understand the system. A prototype is more concrete and easier to visualise than written statements of requirements. This makes it more possible for users, and other stakeholders, to identify problems and suggest improvements (Bray, 2002, Kotonya & Sommerville, 1998; Sutcliffe, 2002).

In Kotonya and Sommerville’s (1998) and Bray’s (2002) view of validation, it involves checking the final draft of a requirements document in which all system requirements should be included. Inconsistencies and incompleteness are identified and resolved. The primary purpose of requirements validation is to answer “have we got the requirements right?” (Kotonya & Sommerville, 1998, p 88). They further claim that validation is also a final assurance that the stakeholder needs are met. Examples of requirements problems that can be found during validation are requirements conflicts that were not discovered during requirements analysis, ambiguities in requirements due to poorly worded statements, faults in models concerning, for instance, the system or the problem to be solved, or standards that are not fulfilled. The outputs of the validation process are both a list of requirements problems and a list of agreed actions that address the identified problems (Kotonya & Sommerville, 1998). A widespread technique for validating requirements documents is requirements review (Eriksson, 2007; Kotonya & Sommerville, 1998).

Like other RE process activities, requirements validation is iterative, as well as interrelated to and interdependent of other activities. Requirements validation can be regarded as part of decision analysis and decision evaluation. One reflection is that it should be possible to develop decision-centred validation techniques, and this way the quality of RE decision-making can potentially improve.

At the centre of attention during most parts of requirements validation is requirements documents.
2.3.6 Requirements documentation

A large amount of resources is invested in the RE process, which results in a lot of information. The information needs to be documented so that it can be used by different stakeholders. Similar to specification, the term ‘documentation’ has two meanings; an activity or an artefact. Both denotations are used and the textual context shows the current meaning. Requirements documentation should be performed continuously in the RE process so that no important information is lost.

A requirements document can be stored and accessed using different media; either a computer or paper. The media can also be combined, e.g., requirements can be stored in a database and from that a paper-based requirements document can be generated. Paper-based documents are still important. To put it in the words of Hoffmann et al. (2004, p 305) “the days of paperless development are still far away, especially in fields where interaction with suppliers is important.”

An important document is the requirements specification (see section 2.2.3). Eriksson (2007) lists other requirements documents, for example:

- Preliminary study documents – The result of investigations preceding the requirements engineering process
- Vision documents – Describe the vision of the system
- Use cases – Describe the interaction between the actors of the system
- Supplementary specifications – Additional requirements
- Change requests – Describe requests of requirements changes
- Sequence diagrams – Describe communication flows between the system actors
- Function specifications – Requirements that are broken down from the requirement specification
- Screen layout – Describe the screen layout of the system
- Design specifications – Requirements that are broken down from the function specification
- Graphical user interface standards – Company or project-specific guidelines concerning the user interface
- Component specifications – Describe in detail the components of the system

A software requirements document contains, according to Pressman (2000), a detailed functional description, a representation of system behaviour, an indication of performance requirements and design constraints, appropriate validation criteria, and other relevant information. A requirements specification defines the system services in detail. It may function as a contract between the customer and the software developer. A software specification forms the basis for design and implementation. It consists of an abstract description of the software and it can complement the requirements specification (Sommerville, 1995).

The requirements documents should be written and designed in a way that facilitates reading and understanding, for example, structured lists should be used instead of
long, dense texts (Sutcliffe, 2002). The stakeholders of the document need different perspectives of it. Sutcliffe (2002) gives examples of different stakeholder needs. For instance, customers want to be certain that the requirements are beneficial to them, that the requirements address their objectives and meet their organisational needs. Users are interested in the functionality and usability, therefore they need more detailed requirements. Even more detailed requirements are interesting for the software engineers who are going to design and code, based on the requirements, and system testers, who make sure that the system fulfils the performance criteria (Sutcliffe, 2002).

Reading and understanding should be facilitated irrespective of whether the medium is a computer or paper. It is important that the documented information is easy to interpret. If the documents are too cumbersome to read, the stakeholders may avoid reading them. If it is difficult to gain a correct interpretation of the documented information, then there is an increased risk that inappropriate decisions are made and that chosen actions taken are inadequate. It might also be the case that an appropriate way of writing and forming the document on paper might be inappropriate in another medium such as in the computer. The medium should be taken into account when choosing the design of documents, since the interpreted information via the requirements documents affects the outcome of RE decision-making.

The requirements and other important information need to be structured and organised, which is conducted in requirements management.

### 2.3.7 Requirements management

Other important parts of RE is requirements management and requirements change management. Since the requirements are continuously changing, they must be effectively dealt with. There are other significant areas, such as quality assurance in RE, requirements prioritisation, requirements traceability and dependencies, impact analysis, and requirements management tools. We have summarised important areas related to requirements management in Figure 8.
Requirements management is “the structuring and administration of information from elicitation, derivation, analysis, coordination, versioning and tracking of requirements during the complete product lifecycle” (Hoffmann et al., 2004, p 301). Requirements management is an inherent activity in the RE process, and should be “active” through the whole RE process, and thus through the whole systems engineering process. It is, like other activities, interrelated with the other RE process activities.

Proper management of requirements is necessary in order to deal with the changing world of requirements. Dependencies between requirements should be maintained so that proposed requirements changes can be analysed. A lack of high-quality requirements management practices can result in reduced customer satisfaction and increased system development costs (Kotonya & Sommerville, 1998). There can also
be potential requirements that have not yet been included in a requirements specification, and they also need to be stored and managed (Lauesen, 2002). Kotonya and Sommerville include three parts in requirements management; a) management of changes to agreed requirements, b) management of relationships between requirements, and c) management of dependencies between requirements documents and other documents. To cope with this properly, there are some essential practices that need to be conducted, for example, requirements prioritisation and quality assurance (Aurum & Wohlin, 2005).

Implementation of requirements is often costly and takes time. Therefore, not all requirements can be implemented in the system, at least not in the same release. Hence, the requirements have to be prioritised. Berander and Andrews (2005) claim that prioritisation distinguishes the most valuable requirements from the less valuable ones. Requirements prioritisation is considered to be a strategic process in view of the fact that decisions about priorities can constitute the “difference between market gain and market loss” (Berander & Andrews, 2005, p 71). There are two categories of prioritisation techniques, namely a) methods based on quantitatively assigned values, and b) subjective measurement-based negotiation approaches. Important prioritisation aspects are importance, penalty, implementation cost, lead time, internal and external risks, as well as volatility (Berander & Andrews, 2005).

The quality of requirements needs to be ensured. Quality is a complex concept that is dependent on the attributes of the context and organisational perspectives, e.g., user-view, product-view, manufacturing view, and value-based view (Denger & Olsson, 2005). There are also quality attributes for requirements, defined by the IEEE standard. These are correctness, unambiguosity, completeness, and consistency. Requirements should also be ranked for importance and stability as well as be verifiable, modifiable, and traceable. Finally, high quality requirements should be comprehensible, feasible, and have the right level of detail (Denger & Olsson, 2005). Denger and Olsson claim that a quality strategy should be defined in order to address the requirements quality issues.

One reason for addressing requirements management is to tackle the challenges of changing requirements. Requirements change is the “emergence of new requirements or the modification or removal of existing requirements” (Lam & Shankararaman, 1999, p 244). We view requirements change management as a natural part of requirements management. However, as with other activities in the RE process, requirements change management is closely linked to other activities. For example, a requirements change request may call for requirements elicitation if no information is available or it may call for requirements negotiation since it may be the case that not all stakeholders benefit to the same extent from the change request.

There are several types of factors that cause changes to requirements. There are external factors, such as environmental changes, organisational changes, changing customer priorities or evolving user knowledge of the system. There are internal factors, for instance, technical, schedule or cost problems as well as requirements errors, conflicts or inconsistencies (Leffingwell & Widrig, 2000; Kotonya &
Sommerville, 1998). Kotonya and Sommerville (1998) argue that some requirements are more stable than others. Such requirements often represent the fundamental nature of the system. Unstable requirements are called volatile requirements. They are often customer-specific or context-specific, and hence vary between different instantiations of a system (Kotonya & Sommerville, 1998).

Requirements changes begin early in the RE process, and thus requirements change management should be taken into account from the start (Lauesen, 2002). Lauesen describes a basic requirements change cycle that consists of five stages. First, reporting; a requirements issue is reported to the change control board (CCB). Second, analysis; the current problem is analysed along with other concerns, for example, if it is a new demand or a misunderstanding, what the real need is, and what the consequences are. Third, decision; the change request accompanies several decision matters, such as to accept or reject, to include in next release or postpone, or to use other ways of dealing with the problem at hand. Fourth, reply; the decision outcome should be disseminated to relevant stakeholders. Fifth, carry out the decision; take the necessary actions to implement the decisions.

There are some essential practices in requirements change management, such as requirements traceability, requirements dependencies, and impact analysis. Requirements traceability is the “ability to describe and follow the life of a requirement, in both a forwards and backwards direction (i.e., from its origins, through its development and specification, to its subsequent deployment and use, and through periods of on-going refinement and iteration in any of these phases)” (Gotel & Finkelstein, 1994, p 94). According to Dahlstedt and Persson (2005), requirements traceability is accomplished by relating pieces of information to each other. This includes that a) requirements are traced to and from related system components, b) system objectives are traced to and from requirements, c) change proposals are traced to and from requirements, d) a decision is traced to its rationales and assumption, e) test cases to and from requirements, and f) system components and its implementation resource needs are traced to and from requirements. Without traceability information and information about requirements dependencies it is not possible to do impact analysis (Kotonya & Sommerville, 1998).

Dahlstedt and Persson (2005) describe different kinds of requirements interdependency types; a) structural, b) constraining, and c) cost-value interdependencies. Structural interdependencies are hierarchical and cross-structure relationships within a set of requirements. Such interdependencies can be categorised as either a) refined to, b) changed to, or c) similar to. Constraining interdependences are either a requirement that is dependent on the realisation of another requirement or a requirement that is in conflict with another and the two cannot exist or be adequately fulfilled at the same time. Cost-value interdependencies concern the cost of implementation and value of fulfilment of requirements. If a certain requirement is implemented and fulfilled, then the cost or value of another requirement is increased or decreased (Dahlstedt & Persson, 2005).
Requirements changes can have ripple effects on the system and affect, for example, other requirements or organisational goals. The changes have to be feasible, economically sensible, and contribute to customer values. Such investigations are carried out in impact analysis (Lock & Kotonya, 1999). To put it in the words of Bohner and Arnold (1996 in Jönsson & Lindvall, 2005, pp 117-118), impact analysis is “the activity of identifying the potential consequences, including side effects and ripple effects, of a change, or estimating what needs to be modified to accomplish a change before it has been made”. Common strategies for impact analysis are a) analysis of traceability or dependency information, b) slicing techniques in which independent slices of a program are analysed, c) check design specifications and other relevant documentation, and d) consultation of knowledgeable developers. The first two strategies are automatable and the last ones are manual (Jönsson & Lindvall, 2005).

In requirements management, which includes requirements change management, there are many decision challenges. A large number of decisions have to be made, e.g., deciding the priority of requirements, which in turn can aid the decision concerning the order in which the requirements should be implemented. Another decision is deciding whether or not a requirements change request is accepted, and if so deciding when the new requirement should be implemented. Thus, a substantial amount of decision-making activities have to be carried out in requirements management. Several essential practices presented above support decision-making, for example, predicting the potential consequences of a requirements change. If these essential practices are properly included in an RE decision support system, then the RE decision-maker’s abilities can be augmented. By properly, we mean, for example, that user-centred visualisation and interaction possibilities are related to the practices in the system. For example, the RE decision-maker can obtain an overview of dependencies between requirements, change the granularity of the view, and explore side and ripple effects.

The management of requirements can be facilitated by using a requirements management tool. This has not only the potential to store requirements, but can also provide more advanced functionality. Requirements management tools are a type of RE tool.

2.4 Requirements engineering tools

Requirements engineering involves many complex and important activities. The skills and knowledge of the requirements engineers are vital for successful RE work. However, as the size of projects and/or systems increase, then skill and knowledge is not always enough. The requirement engineers possibly need assistance in order to effectively carry out the RE tasks. Such assistance can be partly provided by RE tools. RE tools are defined by Matulevičius (2005, p 61) as “software tools that provide automated assistance during the RE process and support the RE activities”. Often, the tools for RE are termed requirements management tools, (see for example Carvallo et al., 2005; Hoffmann et al., 2004; INCOSE; Lang & Duggan, 2001). Our viewpoint is that not only the management of requirements should be supported.
Also other RE activities and tasks need support, such as generation of ideas of innovative solutions for the next generation of the system, problem solving with regard to requirements error reports, and dissemination of requirements information. Consequently, we use the term RE tools, instead of requirements management tools, in this thesis. According to INCOSE (2004), there are different types of RE tools which are depicted in Figure 9.

A requirements generation tool produces lower level requirements in a systematic way. The generation is based, for instance, on design constraints and results from system simulations (INCOSE, 2004). It is not obvious from INCOSE’s description, if the generation is automatic or supported and manually performed. However, we guess that both are possible to some extent, although we argue that human intervention is important. There are important aspects that are implicit and knowledge-based, which calls for a person. INCOSE (2004) describes a requirements classification tool as enabling classifications of requirements with the purpose of facilitating scheduling and tracking of requirements analysis activities. Requirements capture tools compile information from several text sources. They aid in finding relationships in the documents. A requirements identification tool separates requirements in a set of information from superfluous information. A requirements traceability tool makes it possible to manage links between requirements and other artefacts, such as models, requirements changes, and information sources (INCOSE, 2004). This categorisation shows that not all aspects of RE are supported by tools. For example, there is no requirements negotiation tool or requirements validation tool. Perhaps such tools exist, though not explicitly part of the categorisation of RE tools, which indicates that there is room for improvement in this area.

A problem with the RE tool categorisation of INCOSE is that the functionality of a certain RE tool can be broad and the tool can be classified in several RE tool categories (Matulevičius, 2005). This makes the classification more difficult to use,
which decreases its value. Another problem with the categorisation is that many RE tools on the market are available as COTS components (Carvallo et al., 2005; Heindl et al., 2006). Thus, depending on what components the purchaser chooses, the RE tool provides different functionality. Hence, it can be categorised in different ways.

RE tools are evolving towards integration with other tools used in the development process, and they also progress towards product life-cycle management (Ebert & Wieringa, 2005). However, most RE tools are based on a requirements database, which store requirements and related documents. This facilitates gaining an overview, organising, and finding requirements (Eriksson, 2007). Eriksson states that several RE tools can manage requirements, error reports, as well as test cases. They can be linked to each other and their relationships can be displayed in traceability matrices. Some tools support prioritisation and requirements checking (Eriksson, 2007).

There are many lists of characteristics, requirements, and improvement suggestions for RE tools in the literature (see e.g. Ebert & Wieringa, 2005; Eriksson, 2007; Hoffman et al., 2004; INCOSE; Lang & Duggan, 2001; Matulevičius, 2004). Examples of the content of the lists are that the RE tool must support base-lining and configuration management, be user-friendly, support standard systems modelling techniques and notations, allow the user to freely define a requirements management model, improve facilities for the geographically distributed collaborative work, and inter-tool communications. Our impression is that there is a large number of relevant and important characteristics, requirement, and improvement suggestions. Interestingly enough, none of these lists include characteristics or requirements for RE decision support.

RE tools have advantages compared to office and modelling tools (e.g. Word, Excel, and Visio). A case study demonstrated that the quality of requirements documents was higher, when an RE tool had been used compared to when standard office and modelling tools were used (Matulevičius, 2004). More specifically, as described by Eriksson (2007), RE tools have dedicated functionality for efficient management of requirements. It is possible to depict an overview of relationships between requirements. RE tools enable checking and approval of requirements. Neither of this can be done efficiently and smoothly with standard office and modelling tools (Eriksson, 2007). With the help of RE tools, the goals of systematic requirements management can be achieved (Hoffmann et al., 2004). Other advantages are that RE tools force requirements engineers to write and structure the requirements in a uniform way (Eriksson, 2007). RE tools help overcoming the complexity of activities embraced by the RE discipline (Carvallo et al., 2005). Project managers, requirements engineers, as well as other project participants can benefit from proper management of this complexity with an RE tool (Heindl et al., 2006). However, a survey conducted by Matulevičius (2005) demonstrates that mainstream RE practice relies on standard office and modelling tools instead of specialised RE tools.

There can be many potential reasons why RE tools are not used in every RE process. One reason can be that not all companies and all projects can benefit from RE tools.
According to Eriksson (2007) and Hoffmann et al. (2004), a company needs to have a stable and mature work process to utilise RE tools. The requirements have to be written in a standardised way, which calls for an agreed way of working with requirements. The need for an RE tool increases along with several factors: a) when the number of requirements exceeds about 100-150, b) when there are multiple persons involved in the RE work, c) when the project, budget, risks, and participants increase, d) when requirements are frequently reused, as well as e) when the project is conducted in cooperation between several organisations (Eriksson, 2007; Hoffmann et al., 2004). Hence, immature organisations, small projects, projects with few requirements, and/or single organisations can be overwhelmed by the administrative burden caused by an RE tool. It can be more beneficial in such cases to manage the requirements in a more simplistic manner.

Some other reasons for organisations not using RE tools can be the drawbacks and remaining challenges of current RE tools. Many tools lack a user-friendly interface and they have, therefore, received negative feedback from engineers (Hoffmann et al., 2004). Hoffmann et al. also mention that the effort sometimes exceeds the benefit, for example, when maintaining traceability. This means that the cost of maintaining traceability links goes beyond the gains of always having them up to date. Another problem is the difficulty of making tools from different suppliers works together (Eriksson, 2007). Since RE is an intrinsic part of systems engineering, integration with other tools is important in order to achieve a smooth work process. Lang and Duggan (2001) claim that the social process of software development has not been taken into account in RE tools. The collaborative needs of multidisciplinary and distributed teams are not support enough (Lang & Duggan, 2001). However, as Matulevičius (2005) puts it, the long-term value of an RE tool survey is limited, since their features and qualities continually evolve.

We have argued that the RE process consists of decisions and decision-making activities which are critical for both the system to be as well as the systems engineering process. The RE decision-maker’s abilities and capabilities can be enhanced if appropriate RE decision support is provided. Thus, an important RE tool category should be RE decision support systems (REDSS). For clarification, in this thesis we use the term REDSS to denote a visionary, non-existing tool and the term RE tool represents existing tools. The term REDSS implies that the fundamental concept of such a tool would be decision support. This does not exclude the possibility of implementing decision-supporting functionality and qualities into existing RE tools. Including decision-supporting features in RE tools is desirable.

There is also a lack of descriptions of the types of characteristics an REDSS should have and what requirements we should have of such a system. RE decision support characteristics have the potential to suggest and direct research and development efforts concerning decision support for RE decision-makers.
2.5 Requirements engineering as decision-making

Requirements serve as verbalisation of decisions concerning the functionality and qualities of a system. Thus, the RE process can be viewed as a decision process and requirements can be viewed as decisions (Aurum & Wohlin, 2003; Evans et al., 1997). Requirements engineers can then be regarded as decision-makers.

RE decision-making is complex, has several difficulties, and is of vital importance for both the development process and the system. Therefore, RE decision support can increase the effectiveness and efficiency of RE decision-making.

2.5.1 RE consists of decisions

RE is largely a decision-making process (Aurum & Wohlin, 2003) or in the words of Evans et al. (1997, p 435) “For the engineering of computer-based systems, the term [and the associated process] of ‘requirements’ might well be replaced with the term ‘decisions’ and a decision process”. Stakeholders’ decisions about the quality and functionality of a system to be are expressed in requirements. Other important issues in the RE process, such as organisation, staffing, and planning, are also decided upon. Thus, if poor decisions are made, then RE can fail (Regnell et al., 2001). So, by addressing decision-making improvement in RE, the probability of successful systems engineering increases (Aurum & Wohlin, 2003). Ngo-The and Ruhe (2005) stress that the most successful companies in the future will be those that have an integrated approach for strategic decision-making, requirements management, and road mapping processes. The successful companies will be those who effectively utilise “their intellectual capital generated by the decision-making process and would link this process to the essential supporting information” (Ngo-The & Ruhe, 2005, p 267).

Decisions are made throughout the whole RE process (Aurum & Wohlin, 2003). Aurum and Wohlin compare an RE process model (Macaulay, 1996) with a decision process model (Mintzberg et al., 1976), and they claim that the two have much in common in that their activities have similarities. Using the model of Mintzberg, et al. (1976), micro decisions can be identified. Micro decisions are concerned with the decision-maker level, i.e. how they actually carry out decision-making. Macro decisions, on the other hand, focus on management activities of an organisational level. Micro and macro decisions are mutually dependent and intertwined. Both types are present in RE (Aurum & Wohlin, 2003). Macro decisions can be categorised as belonging to three levels in an organisation: strategic planning, management control, and operational control (Anthony, 1965). In the RE process the decision matter at the strategic level mainly concerns organisational considerations, such as the consistency of requirements with the product strategy or business goals. Tactical decisions, i.e. management control, focuses on the project level, for example, human resource planning. The lowest level, operational control, involves making decisions on realisation issues and decisions on quality, classification, and properties of requirements (Aurum & Wohlin, 2003; Regnell et al., 2001). Examples of decisions that need to be made in RE are (Aurum & Wohlin, 2003; Ruhe, 2003b):
• Which functional and non-functional requirements should be selected in relation to given time and budget constraints?
• How should the requirements be organised?
• How should the requirements be classified?
• What is the importance of the requirement?
• Who are the requirements’ stakeholders?
• What is the priority of the requirements?
• How does the requirement depend on other requirements?

Such decision-making is not always straightforward and there are several challenges facing the decision-makers in the RE process.

2.5.2 Difficulties of decision-making in RE

There are difficulties in RE decision-making, for instance, that it is a knowledge-intensive activity, and that human decision-makers in general have cognitive limitations (Aurum & Wohlin, 2003). However, there are several other obstacles that the RE decision-makers have to deal with. We can use Orasanu’s and Connolly’s (1993) list of eight factors that characterise decision-making in natural settings (see page 66) to structure difficulties of RE decision-making. The factors are a) ill-structured problems, b) uncertain, dynamic environments, c) shifting, ill-defined, or competing goals or values, d) action and feedback loops, e) time stress, f) high stakes, g) multiple players, and h) organisational goals and norms.

Ill-structured problems that are taken care of in a decision-making process seldom reveal themselves in a clear and unambiguous way (Orasanu & Connolly, 1993). This can be said both for software engineering in general (Ruhe, 2003b) and not least in RE (Aurum & Wohlin, 2003). According to Ruhe (2003b), the current state-of-practice shows that the decision problems of software engineering are neither well described and understood, nor are the impact and consequences of decisions comprehended enough.

Uncertain, dynamic environments involve poor data quality and environments that change during the decision-making process (Orasanu & Connolly, 1993). Uncertainty and incompleteness of information are conditions that RE decision-makers often have to deal with. The information, on which the decisions are based, can be fuzzy, inconsistent, or incorrect (Ruhe, 2003b). Another difficulty is when the number of system parts, interfaces, and thus design options increase. Then the decision space can be large and complex (Evans et al., 1997; Ruhe, 2003b). Finally, the environment of RE is constantly changing, which is also a challenge that has to be managed (Ruhe, 2003b).

Shifting, ill-defined, or competing goals or values may cause conflicts and thereby make tradeoffs necessary (Orasanu & Connolly, 1993). Many goals of system development and evolution are conflicting. Examples of shifting goals are usability, security, reliability, maintainability, efficiency, portability, and time-to-market. Trade-offs between them is often necessary (Ruhe, 2003b). There are also explicit and implicit
constraints that need to be taken into account, for instance, constraints related to effort, time, and quality (Ruhe, 2003b).

*Action and feedback loops* are part of the decision process (Orasanu & Connolly, 1993). The decision processes of RE concern system planning, development, and evolution, which is a continuous process that involves problem solving and decision-making. The loops of the process should provide control and understanding of the problem at hand. The decision process has iterations with several levels and review points (Regnell et al., 2001; Ruhe, 2003b). An example of a loop problem in current state-of-practice, according to Ruhe (2003b), is that violations of main constraints are often discovered too late.

*Time stress* is often found to be a part of a decision-making setting, which causes pressure on the decision-maker (Orasanu & Connolly, 1993). The same is often true in RE as well. In current state-of-practice, decisions are often made at the last moment and in an ad hoc manner (Ruhe, 2003b).

*High stakes* are involved when the outcome is of significance (Orasanu & Connolly, 1993). The quality of the software is highly important for market success in all high-tech and service domains. Additionally, the cost of systems is significant (Ruhe, 2003b). Since requirements direct the functionality and qualities of the system to be, as well as the work effort in systems engineering, decision-making in RE has high stakes.

*Multiple players* mean that more that one person is actively involved in the decision process (Orasanu & Connolly, 1993). An essential characteristic of RE is that multiple stakeholders are involved in the process. They add complexity with their different expectations, interests, perspectives, and constraints (Aurum & Wohlin, 2003; Ruhe, 2003b). This calls for negotiations. Misunderstandings between the stakeholders can lead to costly decision revisions (Regnell et al., 2001). According to Ruhe (2003b), a problem in the current state-of-practice of software engineering is that the perspectives of all stakeholders, who may have conflicting interests, are often not taken into account when making decisions.

*Organisational goals and norms* mean that the values and goals used in decision-making are normally part of a broader context, i.e. the tasks carried out in an organisation. Not only personal preferences are used when a decision is made (Orasanu & Connolly, 1993). Of course, RE decision-making is also bound to the organisational context, with its norms, structures, and procedures (Aurum & Wohlin, 2003).

Thus, there are many challenges facing RE decision-makers. These difficulties need to be reduced and the decision-makers need proper support. Consequently, there are many problems to address within research of RE decision-making and RE decision support.
2.5.3 Research within the field of RE decision-making and RE decision support

Research within the field of RE decision-making and RE decision support is in its infancy (Ngo-The & Ruhe, 2005), although there has been and is some research conducted in the field (for an extensive list, see Ngo-The & Ruhe, 2005). Ngo-The and Ruhe discuss that decision problems in RE can be seen from two perspectives; a requirement-centric perspective and an activity-centric perspective. The first one, the requirement-centric perspective, is often the viewpoint of software engineering researchers, whose main concerns are the contexts directly related to requirements. Their decision problems begin with the requirements. The second one, the activity-centric perspective, is often the position of decision theory researchers who apply their theories in the RE domain. Their decision problems are identified in the RE process and the software engineering process. They focus on a broader context and include other aspects, such as maturity of organisations and availability of information. The two perspectives are not mutually exclusive, but the focus and order of importance are different (Ngo-The & Ruhe, 2005). Our perspective is an activity-centric perspective, since we primarily focus on decision-making activities of requirements engineers and the problems and difficulties they experience. Based on the nature of the activities we suggest characteristics of a visionary RE decision support system.

A major challenge of this field is to describe and comprehend RE decision-making. When we have substantial knowledge about this, then it is possible to effectively improve and support RE decision-making. Thus, more theoretical and empirical research is needed (Ngo-The & Ruhe, 2005). To gain insights and guide such research, decision-making theories and models of decision processes can be used. This way we can understand the nature of RE decision-making activities (Aurum & Wohlin, 2003; Regnell et al., 2001). A research agenda for the field of RE decision-making can be derived from Aurum and Wohlin (2003), Ngo-The and Ruhe (2005), as well as Regnell et al. (2001). We need to:

- Perform empirical studies of RE decision-making in a comprehensive and focused way.
- Identify and study the decision problems in the RE process.
- Identify the decision types involved in each RE phase, as well as the meaningful actions or options that each decision-maker carries out for each decision type.
- Develop approaches that properly address requirements decision problems.
- Identify the information type (or knowledge) needed at each phase.
- Examine how non-technical issues, e.g., political, social, organisational, and cultural issues, have an effect on RE decision-making.
- Identify which type of stakeholders participates in each RE activity and accordingly consider specific decision aids for each type of stakeholder.
- Understand the group problem solving processes of RE.
- Validate the impact of improved RE decision-making on the system as well as the systems engineering process.
Based on knowledge about RE decision-making, decision support and decision-making improvements should be suggested. Developing support for RE decision-making is a major issue for RE research (Regnell et al., 2001). “The tremendous impact of software on products and services makes SEDS [software engineering decision support] a critical activity” (Ruhe, 2003b, p 143). Ruhe adds that decision support is needed throughout the whole life cycle.

However, Ngo-The and Ruhe (2005) argue that RE decision support should not strive for optimality. Many decision situations in RE are not simple enough to enable an absolute optimal solution. The decision problems are “wicked” and trade-offs, uncertainty, and judgements are necessary. Instead, the provided support should augment the decision-making capacity of the human decision-maker. The strengths of humans and the computational power should be combined. Humans, for example, have a good capability in handling soft and implicit constraints and objectives, while computational models have a high capacity, such as memory space and computational complexity, where the human cognitive abilities are limited, (Ngo-The & Ruhe, 2005). A research agenda and suggestions for what needs to be supported are provided by Aurum and Wohlin (2003), Ngo-The and Ruhe (2005), Regnell et al. (2001) and Ruhe (2003b). We should:

- Develop empirically based guidelines to support decision-making
- Focus on both improved decision quality and improved cost-benefit decision-making, i.e. both effectiveness and efficiency
- Keep track of RE decisions and their effect on the software product, i.e. record decisions, their rationale, and facilitate traceability
- Emphasise decision support for RE decisions during uncertainty. Approaches from other disciplines can be used, e.g., probability theory
- Provide decision support tools for both development teams and project managers
- Support decision-making and problem solving of groups of stakeholders
- Support requirements negotiations so that all stakeholders’ interests are taken into account
- Make generation and evaluation of alternative solutions possible
- Make better reactions for changes possible
- Facilitate more transparent and robust decisions that can be understood by the stakeholders

Thus, the field of RE decision-making and RE decision support is still immature and a coherent body of knowledge does not yet exist. The field has a lot of potential and can hopefully contribute significantly to RE practice in the future, thereby having a positive impact on the quality of systems and their development processes.

2.6 Chapter summary and reflections

In this chapter, we present the state-of-the-art of requirements engineering (RE). RE is an inherent part of systems engineering and an important purpose is to specify an
agreed set of systems requirements. This is conducted in several interrelated, interdependent and iterative activities. We need to generate the requirements and collect relevant information, which are conducted in requirements elicitation. The requirements need to be investigated, so that the set of requirements are complete and consistent. This is performed in the activity called requirements analysis. The requirements have multiple stakeholders, all with their own opinions and needs. Therefore, the activity termed requirement negotiation is necessary in order to make trade-offs and reach an agreement on which requirements will be implemented. The requirements must also be validated in order to ensure the “right” requirements have been included, as well as checking the quality of the requirements documents used in other parts of the development process. The requirements have to be managed, i.e. structured and controlled.

A requirement can be regarded as a representation of a specific decision concerning a particular aspect of a system. These decisions govern the development process as well as the character of the product. If an inappropriate decision is made, then both the development process and the product can be negatively affected. If we claim that the RE process is a decision-making process, then we can argue that the RE process can also be improved in terms of better decision-making. This means the potential increase of positive consequences of implemented decisions. Positive consequences can, for example, be increased user satisfaction, reduced implementation costs, or products which are better adjusted to the market. It also means that the decision-making activities within the process are conducted in a cost-efficient way.

To improve decision-making, we need to identify and understand the obstacles of RE decision-making. Otherwise, we will not know which challenges to address or how to support RE decision-making. This means that it is essential to gain a comprehensive understanding of the RE decision-makers’ decision situation, e.g., which decisions are actually made; what factors affect RE decision-makers; what decision-making activities are carried out; and which decision processes exist? This understanding enables a definition of what kind of decision support RE decision-makers need and what should constitute such support?

There are inadequate descriptions of the kinds of characteristics such an RE decision support system should have. RE decision support system characteristics have the potential to suggest and direct research and development efforts concerning decision support for RE decision-makers.

The variants of RE processes cause differences in how the activities are carried out. Which aspects of an RE activity that attract extra attention is dependent on the type of RE process. This implies that the need for RE decision support may also be divergent. This does not necessarily mean that a certain RE decision support should be dedicated to a particular variant of RE. However, flexibility in how to use the support is probably of vital importance.

The field of RE decision-making and RE decision support is still immature and a coherent body of knowledge does not yet exist. The field has a lot of potential and
can hopefully contribute significantly to RE practice in the future, thereby having a positive impact on the quality of systems and their development processes.

This concludes the first step towards a human-centred RE decision support system, in which we have walked through the state-of-the-art of the RE field. The next step is to elaborate important aspects of human decision-making.
3  Decision-making – An analysis of theories in use

In this chapter, theories of human decision-making are presented. In order to support decision-makers, e.g., RE decision-makers, efficiently, there is a need to understand the decision-makers’ situation. First, the concept, decision, is elaborated, since decisions are the central part of decision-making, and a decision matter can have different characteristics. Second, the decision-maker is in focus, because they carry out decision-making activities within a decision process. Decision-making activities and decision processes are described in the third section of this chapter. There are many decision-making theories that, for example, concern individual decision-making, group decision-making, and organisational decision-making. The chapter is summarised and some reflections are provided in the last section.

3.1  Decisions

In this section, we first define central concepts such as decision, decision-making, decision-maker, and decision process. Then we describe how decisions can be characterised along different dimensions.

3.1.1  Definitions of central concepts

Decision is a commonly used word. Large amounts of literature have been written on how people make decisions, how we should make decisions, consequences of decisions et cetera. However, the definition of the concept, decision, is still ambiguous. The reason for this could be that the term is taken for granted; everybody “knows” what a decision is. If so, there can be a risk that we interpret the term, decision, differently. Two definitions of the term are a) a decision is a “specific commitment to action” (Mintzberg et al., 1976, p 246) and b) decision is a “reasoned choice among alternatives” (Mallach, 1994, p 28). However, these definitions focus on different aspects. Mallach (1994) focuses on the choice, which in this case must be preceded by an evaluation of alternatives. This means that decisions are regarded as two steps within a decision process. The decision process is often considered to include more steps, which makes defining decisions as the last parts of a decision process somewhat awkward. In the definition of Mintzberg et al. (1976) decisions are regarded as a consequence, and it also implies that some type of action is always the consequence. While it is reasonable to consider a decision the result of a decision process, it does not necessarily cause action. The result can be the decision not to act. In this thesis the central concepts are defined as follows:

- A decision has two meanings; decision matter and decision outcome. A decision matter is the issue that is dealt with in the decision process. The decision matter becomes a decision outcome when the choice is made. The decision outcome is the chosen alternative that is to be acted upon.
• *Decision-making* is considered to be the mental or physical activities done by a decision-maker when dealing with decisions.
• A *decision-maker* is a person who carries out decision-making activities, alone or together with others, but is not necessarily the one who authorises the decision.
• A *decision process* is viewed as a number of phases or steps related to each other that consist of decision-making activities.

These related concepts are shown in Figure 10. A decision-maker deals with decision matters when carrying out decision-making activities. These activities are related to each other, and form the decision process. The decision matter becomes a decision outcome that is going to be acted upon.

![Diagram showing the relation between the central concepts](image_url)

**Figure 10, The relation between the central concepts**

Requirements engineering (RE) can be viewed as a decision process as elaborated in the previous chapter (see section 2.5). Therefore, the general definitions of decision-making can be inherited by RE. Hence, RE decision matters comprise the issues that are dealt with in the RE decision process, which results in an RE decision outcome. The RE decision outcome is what is to be acted upon. RE decision-making is the mental or physical activities done by an RE decision-maker when dealing with decisions. The RE decision-making activities are related to each other in an RE decision process. An RE decision-maker is a person who carries out the decision-making activities, alone or together with others, but is not necessarily the one who authorises the decision.

Decisions can be very different from each other. Some decisions are easy to make, while others are difficult. Some have an insignificant effect, others have a massive impact. Decisions can be characterised along several dimensions.
3.1.2 To characterise decisions

Decisions can be characterised in several ways. According to Mintzberg et al. (1976) there are three ways of classifying decisions, a) by the stimuli that initiates the decision process, i.e. opportunity, crisis, and problem, b) by solution, i.e. given, ready-made, custom-made, and modified, and c) by its process. Another way of characterising decisions is presented by Holsapple and Whinston (1996): structuredness, negotiation, management level, functional area, and managerial function (see Figure 11). Three of the categories are specifically focused on management and business organisations, which are not the focus of this work. Consequently, they have been generalised to fit decision-making in general. The management level is labelled scope and time frame. The new term for functional area is domain-specific functional area, and managerial function is called general functions. The characteristics are described in the following.

![Figure 11, Ways of characterising decisions (adapted from Holsapple & Whinston, 1996)](image)

**Degree of structuredness**

Simon (1960) suggests that decisions can be termed as programmed and nonprogrammed. Programmed decisions are repetitive and have a defined procedure, while nonprogrammed decisions are to a higher extent novel and unstructured. These are not distinct types of decisions, but are instead at the ends of a continuum (Simon, 1960). Gorry and Scott Morton (1971) introduced new terms for these types, *structured* for programmed and *unstructured* for nonprogrammed. They also introduced the term, *semi structured*, for decisions that are partly structured and partly unstructured. This way of viewing decisions is widely used in decision support system literature (see for example Mallach, 1994; Holsapple & Whinston, 1996; Marakas, 1999; Power, 2002).

**Scope and time frame**

Scope and time frame is a way if characterising decision that is similar to what Holsapple and Whinston (1996) call management level. The management levels were introduced by Anthony (1965), who categorises decisions as belonging to three levels in an organisation. However, this way of regarding decisions can be used to describe
other decision situations, e.g., in the political arena, in private life, or in military settings. Figure 12 illustrates it is primarily a matter of scope (from wide range to short range) and time frame (from long-term to short-term). The levels are a) strategic decisions, b) tactical decisions, and c) operational decisions.

![Diagram of strategic, tactical, and operational decisions]

**Figure 12. Scope and time frame for strategic, tactical, and operational decisions.**

At the top level there are *strategic decisions*. This level is called strategic planning and is defined as “the process of deciding on objectives of the organization, on changes in these objectives, on the resources used to attain these objectives, and on the policies that are to govern the acquisition, use, and disposition of these resources” (Anthony, 1965, p 16). Decisions at this level affect a major part of the organisation for a long period of time (Mallach, 1994).

The next level below is *tactical decisions*, which is termed by Anthony (1965) as management control. Management control is defined as “the process by which managers assure that resources are obtained and used effectively and efficiently in the accomplishment of the organization’s objectives” (Anthony, 1965, p 17). Mallach (1994) claims that decisions at this level are made in the context of previously made strategic decisions and affect a part of the organisation for a limited period of time.

At the lowest level are *operational decisions*, denoted operational control by Anthony (1965). This is defined as “the process of assuring that specific tasks are carried out effectively and efficiently” (Anthony, 1965, p 18). Decisions at this level have an effect more on a daily basis.

The characteristics of the needed information differ between the levels. The source of information is to a higher extent external at the higher levels and largely internal at the lower levels. The scope of information is wider and the level of aggregation is higher at the top management level (Gorry & Scott Morton, 1971). It would be interesting to compare the statements of Gorry and Scott Morton (1971) to more recent work, and see to what extent they are agreed upon today. However, such a comparison is beyond the scope of this thesis.
Focus with regard to domain-specific functional areas

Decisions are often classified in terms of functional area of management (Holsapple & Whinston, 1996). Laudon and Laudon (2002) present several functional areas: a) sales and marketing function, b) manufacturing and production, c) finance and accounting, and d) human resources. The sales and marketing function is concerned with identifying customers and their needs, selling the organisation’s products, advertising, communicating with customers and so on. The function of manufacturing and production deals with the planning, development, and production of products. The functional area, finance and accounting, is concerned with handling the financial assets. The human resources function is concerned with the employees and is responsible for attracting, developing, and maintaining the workforce (Laudon & Laudon, 2002).

The disadvantage of using this classification is that it is too narrow. It is too focused on a specific domain, i.e. business organisations. There are other domains, e.g., healthcare, which do not include all of the above mentioned functional areas and might include others. However, if this way of classifying decisions is generalised, it can be valuable. So, depending on the domain, decisions can be classified in functional areas (see Figure 13).

Degree of negotiations

According to Holsapple and Whinston (1996) decisions can also be categorised as negotiated or unilateral. Negotiated decisions are made by multiple participants who must agree. Unilateral decisions may also involve several participants, but there is one individual that actually makes the decision, and the other participants must not agree on the matter (Holsapple & Whinston, 1996).

General function

A decision-maker, e.g., a manager, carries out general activities, which include decisions. Fayol (1984) identified five managerial functions called the POCCC view of management. The functions are general, and not only managerial activities. The functions comprise: a) planning, b) organising, c) coordinating, d) commanding, and e) controlling. Planning is concerned with outlining what to do. Organising concerns structuring human resources, this focuses on responsibility, authority, and expected flow of communication. The coordinating function deals with harmonising activities.
in an organisation, i.e., determining sequence and timing of activities; allocating resources such as time, and priority to things and actions, as well as adapting means to ends. Commanding is done when an organisation starts the process of achieving a goal. The function, controlling, is concerned with ensuring that the plan is followed, verifying the given instructions, and controlling that the appropriate principles are established (Fayol, 1984).

These dimensions through which decisions can be characterised make it possible to create a decision profile. Such a decision profile can deepen the understanding of the nature of the decision matter of interest. In our case, the decisions of interest are RE decisions. To understand the characteristics of RE decisions, we can use the dimensions to depict the profiles of RE decisions. For example, are planning and organising the main concerns of the RE decisions or is it coordinating? Such a decision profile can guide the formulation of characteristics of an RE decision support system (REDSS).

Decisions are made by decision-makers, and it is they who use decision support systems. Thus, we also need to characterise them in order to provide effective and efficient support.

### 3.2 Decision-makers

Decisions are made by decision-makers, sometimes alone and sometimes together with others. However, regardless of the number of people involved, each person has his or her own characteristics.

#### 3.2.1 Classes of decision-makers

Decisions can be made by different types of decision-makers. As can be seen in Figure 14, decisions can be made by either a single decision-maker or by multiple decision-makers together. The single decision-maker can be a person or a machine. However, in this thesis only human decision-makers are of interest. When there are multiple decision-makers involved, they can be a team, a group, or an organisation. Each of these types has its own characteristics.
An individual, human decision-maker has the responsibility for the whole or the major part of a decision process. The outcome of such a process is affected by the person’s unique characteristics, such as knowledge, experience, personality, and cognitive style. Depending on the characteristics, decision-makers benefit from different kinds of decision support (Holsapple & Whinston, 1996; Marakas, 1999). For example, an inexperienced decision-maker may benefit from a decision training system and an experienced decision-maker can be better supported by a decision performance aid.

In cases where there are multiple decision-makers, they are all stakeholders in the decision process. However, they do not necessarily have an equal authority to make a decision, or perhaps none of them has the authority to make a decision on his or her own (Marakas, 1999). In a team of decision-makers there is a manager (or a similar role) and assistants (or similar roles). The manager is the deciding member of the team and the assistants support the manager. The assistants can, for example, be specialists that provide the manager with information, and in this way they influence the outcome of a decision process (Holsapple & Whinston, 1996; Marakas, 1999). In a group of decision-makers there is not a single person who has the authority to make the final decision. Instead, all the members together authorise a decision, which can be made through voting or reaching a consensus. The group may or may not have a leader, but a leader has no formal rights to decide (Holsapple & Whinston, 1996; Marakas, 1999). Decision-makers at an organisational level have similarities to individuals, teams, and group decision-makers, but they have the authority and are charged with the responsibility of making decisions on behalf of the whole organisation (Marakas, 1999).

This classification of decision-makers can be used to clarify the decision situation of RE decision-makers. The need of support is probably different for these classes. Thus, an REDSS can be tailored for individual RE decision-makers, groups of RE decision-makers, or be flexible enough to embrace several decision-maker classes. Nevertheless, regardless of which class of decision-maker we focus on, they all
include individuals and each person has its own personality and decision style that is likely to affect the decision process.

### 3.2.2 Psychological types and decision styles

The impact of personality type on decision-making style is important since the decision support must reflect the decision-making methods of the persons who will use it (Mallach, 1994; Shimizu et al., 2006). Two ways of summarising decision-making styles follow. The first, constructed by Huiit (1992), is based on the psychological types defined by Briggs-Myers and McCaulley (1985), and has its roots in the work of the psychiatrist Carl Jung. The second decision style model is made by Rowe and Boulgarides (1992).

According to Briggs-Myers and McCaulley (1985) the psychological type of a person can be determined by four categories of preferences.

- **Introversion vs. extraversion** is the extent to which an individual prefers to obtain an understanding of the world. Extraverts prefer acting in the world and an introvert prefers pondering about it.
- **Sensing vs. intuition** concerns an individual’s preferred perception process. Sensing type people prefer to take in information through the senses and detailed observations. They find verification important. An intuitive person favours meanings, relationships, and “reading between” the lines.
- **Thinking vs. feeling** is the preferred judgement process. A thinker wants to make conclusions rationally and logically, while a feeling person finds it more important to take emotional aspects into account.
- **Judgement vs. perception** is a life style preference. According to Mallach (1994), people who prefer a judgement process in life, prefer planning and they want things settled and organised. A perception process-preferring individual wants life to be spontaneous and flexible.

These characteristics affect how decision-makers act in a decision process. Huiit (1992) put together psychological types with their characterising decision-making techniques. Extraverts prefer brainstorming in groups, outcome psychodrama, and thinking aloud, while introverts prefer private brainstorming and incubation. A sensing person considers facts (often deliberately overloading) and uses inductive reasoning, i.e. developing rules from specific instances. Intuitive people classify, categorise, reason deductively, i.e. apply rules to specific instances, and visualise. A thinking person also classifies and categorises like the intuitive person. However, a thinking person is more analytical and prefers to use analytical methods. A feeling person shares personal values, pays attention to others’ feelings, and finds clarification important. A judging person prefers evaluation, plus-minus techniques, and backward planning, while a perceiving person prefers brainstorming, provocation techniques, and taking the other’s perspectives (Huiit, 1992).

Rowe and Boulgarides (1992) introduce a decision style model, which is depicted in Figure 15. This model consists of two dimensions, value orientation and cognitive
complexity, which shape four categories of decision styles: directive, analytical, conceptual, and behavioural decision styles.

![Decision style model (adapted from Rowe & Boulgarides, 1992)](image)

Decision-makers with a *directive* decision style have low tolerance for context ambiguity and focus on decisions of a technical nature. They do not want large amounts of information about multiple options. Their problem solving strategy is to use policies and procedures, with their focused nature of thought. In addition, verbal communication is preferred rather than written channels (Rowe & Boulgarides, 1992).

*Analytical* decision-makers can handle large amounts of information compared to directive decision-makers. They can also handle ambiguity well. They have an analytical and insightful problem solving strategy and a logical nature of thought. Written communication is preferred (Rowe & Boulgarides, 1992).

A person with a *conceptual* decision style is highly tolerant of high complexity, but is more oriented towards other individuals than an analytical decision-maker. This person is a creative thinker, but not a doer, and solves problems through intuition and judgement (Rowe & Boulgarides, 1992).

When someone has a *behavioural* decision style he or she focuses on other persons and the organisation. As with the directive decision-maker, the need for structure and low complexity is high. Problems are solved through feelings and instincts. He or she is an emotional thinker (Rowe & Boulgarides, 1992).

Just like all other people, RE decision-makers have their own personalities and decision styles. Consequently, it is important that decision support can be used in a flexible way so that it feels comfortable for each individual. There is otherwise a risk that the REDSS will not suit the decision-makers’ personalities, and perhaps not be used effectively if they find it unpleasant and cumbersome.
Hence, it is valuable to take the decision styles of decision-makers into account when developing a DSS. However, it is even more important to consider how decision-making is actually carried out.

### 3.3 Decision-making

In this section, we first give an overview of decision-making theories. Secondly, theories of problem solving are presented. Thirdly, we describe different models of decision processes. The fourth part contains individual, behavioural decision-making theories, and is followed by examples of group decision-making theories. In the last part of this section, we describe organisational decision-making.

#### 3.3.1 Overview of decision-making theories

Decision-making concerns the activities, mental or physical, carried out by the decision-maker. These activities take place in a decision process. There are many theories of decision-making and models of decision processes. We have developed a conceptual model containing the main research traditions of decision-making and closely related fields (see Figure 16). In sections 3.3.2-3.3.6 the sources used as a foundation for the model are presented.

![Figure 16, Categorisation of decision theories from different research domains](image)

A decision support system aims to support decision-makers and in order to develop efficient and effective DSS we need knowledge of decision-making. One way of increasing our knowledge is to use theories of decision-making, theories of problem solving and models of decision processes.

*Decision-making* is done by a single *individual* or by a *group* of individuals, who might belong to an *organisation* and make decisions for it. The individual, behavioural decision theories are divided into three groups: normative, descriptive, and prescriptive theories.
Problem solving is closely related to decision-making. Often, when a decision is to be made, work has to be done concerning identifying a problem and finding or developing alternative solutions for it. Thus, discussing decision-making without including aspects of problem solving is difficult.

Models of decision processes include activities, viewed as steps or phases, that can or must be performed, and how these are related to each other. Such models can enhance our understanding of the work flow of decision-makers, and might be used, for example, as a base for connecting the right type of information, analysis tools, communication tools, etcetera, to a certain decision-making task when developing decision support. They can also be related to decision-making theories, because it should be possible to connect each theory to one or more activities in a decision process.

Theories of decision-making and problem solving as well as models of decision processes can provide us with directions of what to investigate concerning RE decision-making. They can also be used in order to explain empirical findings of RE decision-making and predict their possible consequences of them. In turn, this can guide our efforts to depict a visionary REDSS.

Most likely, many RE decision matters have a problematic nature which calls for theories of problem solving in order to gain a deeper understanding of them.

3.3.2 Problem solving

Problem solving is a mental activity closely related to decision-making, especially situations where the decision is unstructured and complex. In such cases, the solution to a problem is not obvious and needs problem solving. A problem is a situation where a person’s goal does not agree with the current state and where the person does not know how to reach the goal. Problem solving is the search for a way that leads from the current state to the goal state (Parkin, 2000; Reisberg, 2006).

In this section, six different aspects of human problem solving are explained. First, a presentation of problem solving as searching, second, a description of general problem solving heuristics, and third an introduction to the use of analogies. This is followed by a discussion of problem solving experts, are discussed, the aspect of defining the problem is illustrated, and finally creativity is described.

Problem solving as searching

Newell and Simon (1972) liken problem solving to a search. The search begins from an initial state that should lead to the goal state. The initial state consists of knowledge and available resources. There are also operators and constraints. Operators are the tools and actions that can contribute to approaching the goal state. Constraints make some alternative solutions impossible (Newell & Simon, 1972; Parkin, 2000; Reisberg, 2006).
Parkin (2000) and Reisberg (2006) describe all states that exist between the initial state and the goal state as the problem space. The problem space consists of all the paths to the goal that branch out from the initial state. When problem solving is viewed as a search, it is the search for the path through the problem space that is referred to. Of course, it is impossible to search through all the possible actions that can be taken in the problem space, caused by, for example, the limited capacity of humans’ short term memory. As a consequence, perhaps only one or two alternative paths are evaluated at a time (Parkin, 2000). This means that humans need problem solving heuristics to make the problem space manageable. At the same time, the subset of alternatives that actually consist of a solution to problem should be considered (Parkin, 2000; Reisberg, 2006).

For an RE decision-maker, problem solving is a natural part of the everyday life. Thus, theories of problem solving can be utilised to underpin the descriptions of decision situations of RE decision-makers. The support that could be provided, based on the theory of problem solving as a search, is to make it possible to externalise the search in the problem space and provide memory aid to the RE decision-makers so they are not compelled to rely on their short term memory.

However, humans do not randomly search the problem space. Instead, they use general problem solving heuristics, of which some are powerful and others more uncertain.

**General problem solving heuristics**

Two examples of general problem solving heuristics are the hill-climbing strategy and means-end analysis. Parkin (2000) and Reisberg (2006) explain the first strategy. In the hill-climbing strategy, the problem solver always chooses one that is directed towards the goal for the next step. Humans often rely on this strategy in spite of its limitation. Sometimes, a problem requires taking a step that is not directed at the goal to be able to solve the problem (Parkin, 2000; Reisberg, 2006).

The means-end analysis is described by Parkin (2000) and Reisberg (2006) as a strategy that begins with a comparison between the current state and the goal state. Then, the available means of procedure from the first state to the other are investigated. This problem solving heuristic has two main strengths. Firstly, it makes humans aware of the differences in the present case and it makes clear what is required to solve the problem. Secondly, the strategy often causes the problem solver to break down the problem into sub-problems. This makes the means-end analysis powerful. By addressing the goals of the sub-problems, the work of finding solutions is made easier to grasp and leads to a solution of the main problem (Parkin, 2000; Reisberg, 2006).

The RE decision-makers probably use both problem solving strategies. In particular, they most likely use the means-end analysis since, for example, a known RE practice is to break down higher (e.g. goal) level requirements into lower level and more concrete requirements. Thus, an REDSS can support the RE decision-maker by
facilitating breaking down problems into sub-problems. It should also be possible to see the relations between the sub-problems and the main problem.

It is also important for RE decision-makers to use their knowledge and experiences in problem solving. This can be reflected in the use of analogies between similar problems.

**The use of analogies**

All solutions to problems are not necessarily innovative. Instead, humans can also make use of previous solutions to similar problems. With the help of analogies, unknown solutions can be derived from known ones (Parkin, 2000; Reisberg, 2006). Unfortunately, as Parkin and Reisberg emphasise, humans have difficulties using analogies. In particular, while the spontaneous use of analogies is rare, humans are more disposed to use them if they are instructed to do so.

Parkin (2000) and Reisberg (2006) mean that humans need to go beyond the superficial characteristics of a problem – the surface structures – and instead focus on the principals that control the problem – the deep structures. Then, the possibilities for a problem solver to create and understand analogies increase. This means that the use of analogies is promoted if the deep structures are attended to (Parkin, 2000; Reisberg, 2006). Reisberg (2006) stresses that if humans are exposed to the right kind of training problems and are properly educated, then the chances of them identifying the deep structures of problems by themselves increase. This can make their problem solving more effective.

It can be concluded from this aspect of problem solving that more education and experience increase the possibilities of using analogies in an effective way, thereby increasing the chances of better problem solving and decision-making. Of course, this is not a remarkable conclusion. It is a higher probability that an expert is a better problem solver than a novice.

**Problem solving experts**

In order to be solved, many problems that face humans require knowledge. For example, the use of analogies is not a general human ability; instead it depends on domain-specific knowledge (Parkin, 2000; Reisberg, 2006). Reisberg (2006) means that problem solving experts in general are superior at making use of different problem solving strategies since they have a better understanding of the deep structures of the problems. This does not only support the use of analogies, but also the ability to break down a problem into sub-problems. According to Klein and Methlie (1990) and Zachary and Ryder (1997) experts have several characteristics that differ from novices’:

- **Performance:** Experts are more efficient and can use their knowledge to draw correct conclusions, even if the available information is not complete.
- **Discrimination:** An expert does not start solving a problem through specifying all possible causes. Instead, experts quickly distinguish relevant information from irrelevant and possible causes from impossible causes.
• **Pattern recognition:** An expert can recognise the pattern of a certain type of problem, and connect a possible solution to it.

• **Domain knowledge:** Expertise consists mainly of domain knowledge, rather than general problem solving strategies and methods. An expert knows the interrelationships between concepts, i.e. causal relationships. The domain knowledge of a novice primarily consists of facts and basic concepts.

• **Unconscious knowledge:** The expert is not consciously aware of much of the knowledge related to the expertise. Often, experts have difficulties verbalising their knowledge. They must be contextually stimulated in order to make their knowledge conscious.

• **Theoretical and experiential knowledge:** The theoretical knowledge of an expert is both conceptual and analytical, while experiential knowledge is gained through training and practice.

• **Goals:** An expert has the ability to chunk sub-goals and has a global focus. Novices, on the other hand, have a more local focus and treat sub-goals in a more sequential way.

• **Methods:** Experts have the possibility of performing case-based and more intuitive problem solving. They use strong, domain-dependent methods. Novices tend to be more analytical in their problem solving and use weak general methods.

RE decision-makers can be both novices and experts. This means that it is important that REDSS is flexible enough to support different levels of experience. The evolution from novice to expert can be especially important to consider when decision training is a purpose of the REDSS.

A general difference between experts and novices is probably their ability to define undefined problems.

**Defining the problem**

Many problems are not served prepacked with clear frames, goals and starting points. Such problems are, in the words of Reisberg (2006), undefined. There are several ways of dealing with undefined problems. One way is to create sub-problems and thereby have the possibility of solving the main problem by solving the sub-problems one at a time. Another way is to establish a structure of the problem by adding more assumptions or limitations. This way, a better defined problem gradually evolves with clearer goals and a manageable set of operators and alternatives. However, not even well defined problems are unambiguous, but open for interpretations (Reisberg, 2006).

Parkin (2000) and Reisberg (2006) describe negative tendencies of humans in relation to defining problems. A negative tendency is called functional fixedness. Humans are sometimes biased to rigidity concerning the function of an object. If functional fixedness occurs, then it is difficult to find, in particular, creative solutions to the problem (Parkin, 2000; Reisberg, 2006). Another potential obstacle is *Einstellung*, which is the German word for attitude. Humans are biased in acquiring a deadlocked attitude or a one-eyed perspective regarding how to approach a problem.
This leads to a tendency to tackle all problems in the same way. Of course, it can be effective to use a certain strategy that experience has shown to be appropriate. However, it can, at the same time, be an obstacle to developing better problem solving strategies (Reisberg, 2006).

RE decision-makers also face undefined problems that demand work to understand and solve. If REDSS can facilitate defining a problem and avoid negative human problem solving tendencies, then RE decision-making can be more effective. Another skill that is important in RE decision-making is creativity.

Creativity

A capability that is significant for problem solving is creative capacity. A classical description of the creative process was proposed by Wallas (1945), who claims that complex problem solving consists of four stages:

- **Preparation** includes collecting information and attempting to make initial solutions. This stage is often experienced as arduous and frustrating.
- **Incubation** is when the individual does not think consciously of the problem. Instead, the problem is unconsciously treated. This stage leads to the next one.
- **Illumination** in which new ideas and insights flourish and solutions suddenly appear.
- **Verification** is when the details of the ideas increase and the solutions are ensured.

Reisberg (2006) criticises Wallas (1945) creative process. There are creative processes that evidently do not have all the steps. Although, many creative processes include these steps, they do it in a more complex and iterative way. Another criticism is that it has not been possible to prove the incubation effect and it can, at best, be viewed as unreliable. Concerning the illumination step, it is more a discovery of a new way of tackling the problem and not a solution of it. The new approach can be valuable, but it can also be an impasse (Reisberg, 2006).

To be creative is essential for an RE decision-maker, since it is sometimes important to, for instance, create innovative features of a system. The creative process can be supported and enhanced by a visionary REDSS.

Problem solving can be one of many decision-making activities in a decision process. There are several models available that can be used to describe decision processes.

### 3.3.3 Models of decision processes

A decision can be preceded by a decision process. A decision process can continue for years or be much shorter. Sometimes a decision must be made here and now, for example, when an accident has just occurred. In such cases, there may be no time to consciously consider the alternatives. The decision is rather a reaction to the current situation. In other cases, a decision process can continue for some time, which allows
the decision-maker to actively and consciously work towards solving a problem and making a decision.

A decision process can be viewed as a “set of actions and dynamic factors that begins with the identification of a stimulus for action and ends with the specific commitment to action” (Mintzberg et al., 1976, p 246). There are many models of decision processes, e.g., Fischoff and Johnson’s (1997) four-step scheme of decision-making and Klein’s (1993) recognition-primed decision model. Three decision processes are presented here; Simon’s (1960), Power’s (2002), and Mintzberg’s et al. (1976). Simon’s (1960) fairly simple model is often adopted in the decision support system literature (see for example Silver, 1991; Mallach, 1994; Marakas, 1999; Power, 2002). The model of Mintzberg et al. is the most complex of the three. It is a thorough model that includes many activities, is highly iterative, and has a flexible view on decision processes. It is flexible in the sense that there are several paths through the process. Both Simon’s (1960) and Mintzberg’s et al. (1976) models were developed some years ago. Therefore, they are compared to a more recent model presented by Power (2002). Fischoff and Johnson’s model (1997) is similar to Simon’s (1960) and for that reason is not described here. Klein’s (1993) RPD model focuses on how experienced decision-makers make rapid “here and now” decisions, when a decision-maker reacts to a situation. That kind of reaction-oriented decision process is not relevant for the topic of this thesis.

Simon (1960) suggests a decision process consisting of three phases. It can be seen as an iterative process where each phase in itself is a complex decision-making process. These phases are the intelligence, design, and choice phases (see Figure 17). According to Sprague and Carlson (1982) a fourth phase, implementation, can be added. Intelligence is the search for and formulation of the problem that calls for a decision. Design is the phase where alternatives, i.e. courses of actions are developed and analysed. Choice is evaluation of alternatives from the design phase, and the selections of one of them. Implementation is when the choice is implemented.

![Figure 17, Decision process according to Simon (1960) and Sprague and Carlson (1982)](image)

Angehrn and Jelassi (1994) argue that Simon’s phase model has provided a sound conceptual basis for building early decision support systems (DSS), and that it was assumed to be valid and useful. However, this model has become an obstacle for the evolution of DSS theory and practice, since alternative perspectives have not been included. By adopting other perspectives or models of human decision-making, the possibilities of supporting decision-makers can be enhanced. Alternative perspectives are, for example, decision-making as a learning process and focus on decision-making biases (Angehrn & Jelassi, 1994).

The main advantage of Simon’s (1960) decision process is the impression of simplicity, which makes it manageable. This impression of simplicity may also be a
disadvantage. Since decision-making is a complex process, it can be treacherous to view it as a plain set of related activities. It can be misleading and make us overlook aspects that may be important to consider.

A general decision process model that includes a larger number of steps is presented by Power (2002). It partly overlaps Simon’s (1960) model and consists of seven steps (see Figure 18).

![Figure 18, A general decision process model (adapted from Power, 2002)](image)

The first step is to define a problem. It is important to define problems well, because if the wrong problem is defined it is not possible to make a correct decision. Organisations are often complex, which makes it harder to recognise a real problem and define it well. It can be difficult to separate real problems from problem symptoms (Power, 2002).

Depending on the decision situation, Power (2002) claims, the decision can be made by a single decision-maker or by a group. Therefore, there is a need to decide who should actually make the decision. Then information is collected so that the decision is not made by hunch and intuition. This step is iteratively related to the fourth step, where the alternatives are then identified and evaluated. Brainstorming and generating ideas are important tasks in this step (Power, 2002). It can be questioned why the phase collecting information is not iteratively related to other phases, especially defining problems, follow up and assessment. These phases probably also need information gathering in order to be carried out properly.

In the fifth step a decision is made, i.e. a commitment to a certain course of action or commitment to passiveness is made (Power, 2002). Decisions trigger actions. The decision is implemented by communicating decisions, plan actions, and track performance. Finally, the last step is follow-up and assessment, where the consequences of decisions are checked. This may lead to the identification of new problems (Power, 2002).
Power’s (2002) model has the advantage over Simon’s (1960) model in that it includes more steps and the risk of overlooking important aspects is hopefully reduced. However, the description of the decision process gives a sequential impression, while decision processes seldom are sequential. It seems reasonable to assume that a decision-maker needs to go back in the process at some point, for instance, to reconsider the defined problem.

A model that offers a more iterative perspective of the decision process is produced by Mintzberg et al. (1976). Their model shows the related activities of strategic decision-making processes and aims to describe the structure of unstructured processes. This framework consists of three phases: identification, development, and selection. Each phase consists of routines, which are shown in Figure 19. The process is iterative. Supporting routines and dynamic factors are also part of the framework, although not shown in the figure.

Figure 19, A model of the strategic decision process (adapted from Mintzberg et al., 1976)

The identification phase consists of two routines. The first, decision recognition, is concerned with recognising an opportunity, problem or crisis that call for a decision activity. Opportunities, e.g., an idea in somebody’s head, and a crisis are often a single stimulus. Problems are often multiple stimuli, which cumulate over time and finally call for a decision. Thus, there are several ways of starting a decision process, not just by identifying problems. The other routine in the identification phase is the diagnosis routine. This is when the decision-maker tries to understand the stimuli and the decision situation.

The second phase, development, is described through the routines search and design. The search routine is targeted at finding ready-made solutions. The design routine
involves development of custom-made solutions or modification of ready-made solutions.

The last phase, selection, can be iterated several times, because a decision is often divided into sub-decisions, where each of which is terminated by a selection. This phase consists of three routines: screen, evaluation-choice, and authorisation. The screen routine is a superficial routine in which decision-makers eliminate non-feasible alternatives, but do not determine what is appropriate. The evaluation-choice routine includes three modes. In the judgement mode, the decision-maker makes choices. In the bargaining mode, a group of decision-makers make choices. In the analysis mode, the actual evaluation is carried out. The authorisation routine is used when the actual decision-maker does not have the authority to formally make the decision.

Mintzberg et al. (1976) include different types of supporting routines in their framework: decision control, decision communication, and political routines. The decision control routine includes planning the approach and allocation of resources needed to handle the decision process. The decision communication routine is the communication carried out in the whole decision process. The political routine, which is important in a strategic decision process, is concerned with both internal and external political activities. Included in the framework are also six groups of dynamic factors that disturb the decision process: interrupts, scheduling delays, timing delays and speedups, feedback delays, comprehension cycles, and failure recycles (Mintzberg et al., 1976).

An advantage of Mintzberg’s et al. (1976) model is that it includes many steps and factors that can support a deeper understanding of the complexity of the decision process. However, it might be too complex for some purposes. If the purpose is to quickly and easily provide a basic understanding of decision-making, for example, when promoting a common understanding in a team with different backgrounds, it is not sufficient. Simon’s (1960) or Power’s (2002) models could then be more appropriate.

There are both similarities and differences between the three models. All three begin with identifying that a decision may be needed. Power (2002) and Simon (1960) have problem-oriented starting points. Power’s (2002) model starts with defining a problem and Simon’s (1960) with the search for and the formulation of problems. Mintzberg et al. (1976), on the other hand, also include opportunities as a type of stimuli that call for a decision. The decision process may be different depending on what calls for a decision. The model of Mintzberg et al. (1976) has a large number of possible paths in a decision process that makes the model more flexible. The models of Power (2002) and Simon (1960) give an impression of being “one way” processes, i.e. mostly sequential and less iterative. It seems probable that a semi-structured or unstructured decision process is highly iterative and that all instances of decision processes do not go through exactly the same activities. Power’s model and Sprague and Carlson’s (1982) addition to Simon’s (1960) model make the decision process longer and more complete than the model of Mintzberg et al. (1976). The first mentioned models include what happens after the actual choice, which makes it
possible to discuss how the implementation and follow up of a decision can be supported. Power (2002) has an early phase, decide who should decide, that is not represented in the other two models. The nearest representation is the routine authorisation in the model of Mintzberg et al. (1976). The last two activities in Power’s (2002) model complement Mintzberg’s et al. (1976) model. In this thesis these models are put together in order to describe decision processes (see Figure 25 on page 110).

The models of decision processes can be used for recreating the actual RE decision processes. Such a model can draw our attention to several different aspects of the RE decision processes and identify the actual RE decision-making activities that take place. The model can guide us when to draw the relations of the activities. There are many RE decision-making activities that relate to each other. A visionary REDSS should not just support one RE decision activity, but also the flow of activities in the RE decision processes.

However, identifying the decision processes is not enough when developing an effective and efficient REDSS. We also need an understanding of the decision-making behaviour of individuals.

3.3.4 Individual, behavioural decision-making

Theories about decision-making can be seen as normative, descriptive, or prescriptive (Bell et al., 1988). Descriptive decision theories aim to describe how decisions are actually made. Normative decision theories show how decisions ought to be made, and prescriptive decision theories are concerned with how people can be helped and trained to make better decisions (Bell et al., 1988). More precisely, they are viewed as follows:

- **Normative** theories state how to choose the optimal alternative, where optimal is defined in a quantitative way.
- **Descriptive** theories describe how decisions are actually made.
- **Prescriptive** theories give guidance on how to act within a decision process.

Klein and Methlie (1990) stress that both descriptive and normative decision theories can be used to gain understanding of decision-making in order to recommend better methods and offer advice so that the decision process can be improved. An important aim of decision support is to increase effectiveness in decision-making. To be able to reach better decisions, there is a need to understand why a certain support increases the quality of a decision. A descriptive approach can, for example, be a good starting point for the improvement and development of prescriptive theories. The normative approach can explain why a supported decision is better than an unsupported one. Decision-makers do not master normative principles by themselves. Therefore, the designer of decision support should use design methodologies that include such principles (Klein & Methlie, 1990).
From this account it follows that all types of individual, behavioural decision-making theories can be used when supporting decision-makers. In the following, all three types of theories are described.

**Normative theories**

Normative theories are the result of research within the discipline of Classical Decision-Making (CDM). CDM has its roots in the work of Daniel Bernoulli, who in 1738 published an important work called “Exposition of a new theory of the measurement of risk” (originally published in Latin) (Lipshitz et al., 2001; Plous, 1993).

Normative theories are based on a rationality paradigm. Normative models define conditions for perfect utilisation. The decision-maker is seen as “the economic man”, who acts rationally, calculates the consequences of each alternative, ranking the consequences and finally makes the optimal decision, i.e. maximises utility (Klein & Methlie, 1990). An optimal decision is the best option, measured in a quantitative way (Edwards & Fasolo, 2001). Within the normative approach, theories concerning how to choose and handle uncertainties are developed based on mathematics (Klein & Methlie, 1990; Edwards & Fasolo, 2001). Although, people are not always good at making rational decisions, but it is possible to formulate theories on how to make a rational decision (Gärdenfors & Sahlin, 1988). Three decision theories that are the “heart of most traditional decision technology” are a) subjective expected utility maximisation, b) Bayes theorem, and c) multi-attribute utility (MAU) (Edwards & Fasolo, 2001, p 581).

*Expected utility theory* is a well-known theory of normative decision-making (Plous, 1993; Lipshitz et al., 2001). Expected utility theory was invented by von Neumann and Morgenstern and first presented in 1944. The purpose of this theory is stated by von Neumann and Morgenstern (1953, p 31) as describing the “mathematically complete principles which define ‘rational behavior’ for the participants in a social economy, and to derive from the general characteristics of that behavior”.

*Bayes theorem* is a formula that calculates probability (Plous, 1993). It includes the decision principle of maximising utility: “In a given decision situation the decision maker should choose the alternative with maximal expected utility (or one of the alternatives with maximal expected utility if there are more than one)” (Gärdenfors & Sahlin, 1988, p 5).

*Multi-attribute utility* (MAU) can be seen as both normative and prescriptive. The borderline between these types of decision theories is not clear. An example of this is the work of Edward and Newman (1982). They present an approach to support evaluation of alternatives called multi-attribute utility technology (MAUT). A MAUT evaluation is based on some key ideas that include both judgement parts and quantitative parts, i.e. it views both subjectivity and objectivity as natural components of the evaluation process (Edward & Newman, 1982). They prescribe seven steps on how to do a MAUT evaluation, where, for example, step two is ‘identify stakeholders’. These steps, together with the idea of subjectivity, make it
possible to view MAUT as a prescriptive theory. However, MAUT also has a
normative side since it includes ways to add quantitative values on qualitative
attributes. These values give an aggregate utility that makes it possible to choose, in a
quantitative sense, the optimal option or options.

If the purpose of an REDSS is to reach more rational decisions, from a measurable
and objective point of view, then using normative decision theories can be
appropriate. They can guide how a decision should be made. However, in order to
develop an effective and efficient REDSS we also need to understand the actual RE
decision-making and which difficulties RE decision-makers face. Thus, normative
theories are unsuitable. Instead, descriptive theories should be used.

Descriptive theories

There are two kinds of descriptive theories of individual decision-making, the
“traditional” theories that are based on laboratory studies, i.e. Judgement and
Decision-Making (JDM), and theories based on studies made in natural
environments, i.e. Naturalistic Decision-Making (NDM).

Judgement and decision-making

JDM theories are focused on how people make choices from a set of alternatives.
They do not, to a large extent, take contextual factors, such as stress, into
consideration. JDM can be traced back to 1954 and the work of Edwards called “The
theory of decision making” and Meehl’s work “Clinical vs. statistical predictions:
theoretical analysis and review of the evidence” (Lipshitz et al., 2001).

Two examples of JDM theories are the satisfying theory (Simon, 1956) and prospect
theory (Kahneman & Tversky, 1979). The satisfying theory claims that decision-makers
do not necessarily choose the optimal alternative, rather an alternative that is good
enough in order to satisfy the needs of a decision-maker (Simon, 1956). The option
that first reaches an acceptable level is chosen (Simon, 1978). The prospect theory
includes two phases. In the first phase, a preliminary analysis of the prospects is
made. In the second phase, the prospects are evaluated and a choice is made.
Decision-makers perceive possible outcomes as gains and losses, which can be
manipulated through the formulation of a prospect. The theory also includes a value
function and a decision weight function (Kahneman & Tversky, 1979).

Another view of what guides human choices is called reason-based choice. According
to this theory, decisions are made when the decision-maker sees compelling reasons
for it. A reason-based choice implies that humans are dependent on finding
persuasive reasons. They also have to judge if the reasons are persuasive enough. An
important factor is emotions. The emotion regret is extra important for decisions,
since humans are strongly driven towards avoiding regret. Therefore, an alternative
that reduces the risk of regret afterwards is strongly appealing. The best cure against
regretting a decision is one that the decision-maker can stand by afterwards. Thus, a
decision that can be justified and explained (Reisberg, 2006).
Other work within this research tradition also often cited in the literature is the work by Amos Tversky and Daniel Kahneman cornering judgement under uncertainty (cited in e.g. Plous, 1993; Eysenck, 1993; and Cohen, 1995). They carried out studies on how decision-makers use heuristics in complex situations and which biases, i.e. patterns of errors, the use of heuristics can lead to. According to Tversky and Kahneman (1974), heuristics are sometimes good because they can reduce time and effort as well as result in an acceptable decision. However, they can also cause negative effects. Two heuristics introduced by Tversky and Kahneman (1974) are the availability heuristic and the representativeness heuristic.

The availability heuristic is used by decision-makers when they assess “the frequency of a class or the probability of an event by the ease with which instances or occurrences can be brought to mind” (Tversky and Kahneman, 1974, p 1127). This means that humans assess the frequency of something based on how easy it comes to mind. This heuristic is also used when the judgement is critical. To make an approximation of the occurrence of an event is called a frequency estimate. A frequency estimate can be significant in determining relations between causes and effects (Parkin, 2000; Reisberg, 2006). However, humans often do not have the needed frequency estimation. Instead, they tend to use availability, i.e. how easily available the information of something is in human memory. If something comes to mind easily, then it is taken for a frequent event. However, this heuristic also leads to biases. Just because an instance of an event comes easily to mind does not necessarily imply that this event will more probably occur than other events (Plous, 1993; Reisberg, 2006). The human memory is constituted in a way that what easily comes to mind is dependent on several factors, not only on the frequency of an event. Humans have a tendency to pay attention to unusual, obscure, or important events and the events paid attention to are easier to remember. Hence, they are more available (Parkin, 2000; Reisberg, 2006).

The representativeness heuristic describes that decision-makers judge probabilities “by the degree to which A is representative of B, that is, by the degree to which A resembles B” (Tversky and Kahneman, 1974, p 1124). This means that humans tend to draw general conclusions from one or a few instances or an occurrence or a population. The origin of this heuristic is a human assumption that all members of a category are fairly homogeneous. This assumption inclines humans inclined to make inferences based on only small numbers of observations. Often, such conclusions are correct. It is possible to transfer characteristics and information from one observation to the whole phenomenon or population. Unfortunately, the representativeness heuristic also leads to untrue conclusions (Parkin, 2000; Reisberg, 2006). Biases resulting from this heuristic are, for example, that decision-makers often believe that a more detailed scenario is more probably representative of some phenomenon, than a more general scenario (Tversky & Kahneman, 1982).

Theories concerning judgment and decision-making can be used for reasoning about RE decision-making and they can be a starting point for how to provide decision support to RE decision-makers. For example, the reason-based choice theory stresses the importance of being able to justify and explain the decision. This is also
important in RE decision-making. RE decision-makers need rationale for their
decisions and these motives should be documented. This implies that the REDSS
should encourage the RE decision-maker to formulate and document the rationales.
It should also be easy to retrieve the documented decisions and their motives. In this
way, we can use the general decision-making theories and utilise them in the field of
RE. There are other research traditions in decision-making, such as naturalistic
decision-making, that can also be valuable for RE decision-making and REDSS.

Naturalistic decision-making

Naturalistic decision-making (NDM) is a relatively new research tradition that began
in the second half of the 1980’s (Klein et al., 1993). The work of Gary Klein called “A
recognition-primed decision (RPD) model of rapid decision making” is the root of
NDM (Lipshitz et al., 2001). It began as a reaction to the studies made of CDM and
JDM. Studies within NDM are made in a natural context, as opposed to laboratory
studies made in traditional decision-making. As Orasanu and Connolly (1993, p 6)
stress, decisions are part of larger tasks and “decision event research in the
laboratory tends to require decisions apart from any meaningful context. In natural
settings, making a decision is not an end in itself.”

Orasanu and Connolly (1993) claim that earlier decision research have not taken into
account the features of the task and the subject’s knowledge and experience relevant
to the task, which they argue NDM does. An example of how the experience level of
decision-makers influences decision-makers is provided by Fischer and Kingma
(2001). They claim that the experience level affects the way decision-makers use
information. It might seem apparent that more experienced persons have more
possibilities to effectively use available information. However, as argued by Fisher
and Kingma (2001), there can be both positive and negative consequences. An
experienced decision-maker may, for instance, have an increased potential to detect
errors in a familiar set of data, but there is also a risk that he or she relies too much
on a feel for the data.

Orasanu and Connolly (1993) list eight factors characterising decision-making in
natural settings.

- **Ill-structured problems:** problems that need to be taken care of in a decision-
  making process seldom reveal themselves in a clear and unambiguous way.
- **Uncertain, dynamic environments** include poor data quality and environments
  that change during the decision-making process.
- **Shifting, ill-defined, or competing goals** or values, which may cause conflicts and
  a need to make tradeoffs.
- **Action and feedback loops** are part of the decision process, since NDM views
decision-making as often containing a series of events.
- **Time stress** is often found to be a part of NDM setting, which causes pressure
  on the decision-maker. Such pressure causes, according to Payne and Bettman
  (1988), that less complicated reasoning strategies might be used.
- **High stakes:** the outcome is of actual significance for the decision-maker.
- **Multiple players**: more that one person is actively involved in the decision process.
- **Organisational goals and norms**: the values and goals used in decision-making are often part of a broader context, i.e., the tasks carried out in an organisation. Not only personal preferences are used when a decision is made.

All these factors affect a decision-maker and the way decision-making is carried out. These factors are related to each other, which can be seen in Figure 20.

![Figure 20, Relations between factors affecting decision-makers](image)

The goals and norms of an organisation can directly guide the decision-maker, for example, in weighing up and comparing potential outcomes of alternatives. However, depending on, for instance, how the goals are formulated and on their content they can affect the factor vague goals. Vague goals can be shifting, ill-defined or competing. If, e.g., the ethical goals of the individual and, e.g., the economical goals of the group are in opposition to each other, the decision-maker has to do trade-offs. Vague goals, together with ill-structured problems and multiple players, can influence the factor, uncertain, dynamic environment. The more participants of a decision process the more can happen during the time a task is carried out. Uncertainty concerning a specific matter, in this case of ill-structured problems, brings uncertainty to the wider context in which the matter is a part. Uncertain and dynamic environments directly influence a decision-maker, who must decide how to deal with this uncertainty, for instance, how to be flexible enough. Since a decision process can include several events and can go on for some time, together with a need to obtain information from the uncertain and dynamic environment, feedback loops are necessary. Two other factors that directly affect the decision-maker are high stakes and time stress. Both increase the pressure on a decision-maker. The dynamic factors presented by Mintzberg et al. (1976) that disturb the decision process probably have an effect on the time stress. The more interruptions, failure delays et cetera the more time pressure on the decision-maker, especially if there is a deadline.
A large number of studies have been conducted within the research domain of NDM; two examples are the work of Brehmer (1992) and Beach (1993). Brehmer (1992) has studied what he calls dynamic decision-making concerning human control of complex systems. Beach (1993) invented the image theory, in which he argues that the decision-maker uses three images when he or she makes a decision. The images are: how things should be, which goals to reach, and images of plans for reaching the goals.

Both JDM and NDM theories are important from an REDSS perspective. JDM provides a narrow-focused view of decision-making. This can be used, for example, to reach an understanding of the importance of how alternatives are presented. NDM provides a broader setting for aspects affecting decision-making. It can constitute a foundation for studying which aspects can be supported and which ones cannot. JDM and NDM theories can also form a basis for the development of prescriptive decision theories.

Prescriptive theories

Prescriptive decision theories can be used in different ways. For example, they can be used by a decision-maker as a checklist for what to do and how to think. In addition, they can be a source for learning how to carry out effective and efficient decision-making. In decision support system development, such theories can be applied as a basis for enhancing the outcome of a decision process. Another way to use prescriptive theories, suggested by Vetschera and Walterscheid (1995), is as an evaluation tool for managerial support systems.

Matheson and Matheson (1998) suggest a prescriptive theory for strategic decision-making. The theory can primarily be seen as an organisational decision-making theory. According to Matheson and Matheson (1998) decision quality can be viewed as a chain consisting of six links. A decision-maker can use these links and the questions related to each link to make better strategic decisions. The links in the chain are:

- **Appropriate frame** includes checking if the right background and context is used for the decision.
- **Creative, doable alternatives** are concerned with finding undiscovered alternatives.
- **Meaningful, reliable information** embraces not only having the right information, but also increases awareness of the limits of the knowledge, i.e. what is not known.
- **Clear values and trade-offs** handle establishment of measurement criteria and how to make rational trade-offs among them.
- **Logically correct reasoning** brings together the four already mentioned links to logically evaluate the alternatives.
- **Commitment to action** connects decision to action. The right people in the organisation must be willing to implement the decision, otherwise the decision is useless.
It is probably possible to use general prescriptive decision-making theories and refine them to RE decision-making. Prescriptive RE decision-making theories can, for example, be used by RE decision-makers in practice. They can guide the RE decision-makers concerning what they should do and how they should do it in order to make effective RE decisions. Prescriptive RE decision-making theories can also, for instance, be used as a foundation for the development of an REDSS. They can provide directions of how the support should be constituted.

Since decision-making is not only conducted by single decision-makers, another layer of decision-making theories can be added, i.e. group decision-making. Group decision-making includes all the problems and difficulties facing individuals, but groups have not only those problems. The group perspective adds other types of difficulties.

3.3.5 Group decision-making

The discipline of group decision-making (GDM) addresses questions concerning what characterises decision-making with multiple participants; what kind of problems they face, and how group decision-making can be improved (Miner, 1992). A decision-making group can be defined as “two or more people who are jointly responsible for detecting a problem, elaborating on the nature of the problem, generating possible solutions, evaluating potential solutions, or formulating strategies for implementing solutions” (DeSanctis & Gallupe, 1987, p 590).

GDM includes, for example, processes and potential problems. Two basic phases in GDM processes are idea generation and idea evaluation. The aim of idea generation is to enhance creativity through reducing negative effects of social interaction. Group norms and pressures can hinder group members from generating or presenting ideas (Miner, 1992). In idea evaluation a choice among the ideas has to be made. Hill (1982) claims that evaluation of alternatives is performed better if it is made by a group compared to what an individual can perform. The same cannot be said of idea generation. It is possible that the performance potentials of individuals in the idea generation phase has been underestimated (Miner, 1992).

There are several potential traps in GDM, such as conformity, groupthink, group polarisation and risky shift, and escalating commitment (Miner, 1992). The conformity phenomenon was, according to Miner (1992), identified by Solomon Asch. This phenomenon raises a social pressure on individuals to change their attitude and behaviour. It has become more important to reach a consensus than make a good decision (Miner, 1992).

Another potential trap, groupthink, is related to conformity. This phenomenon was found by Janis (1982) and occurs when the desire for consensus among the group members is very strong. In such cases, their motivation to think realistically about decision alternatives is reduced. When groupthink occurs it has several symptoms, for example, the illusion of vulnerability, stereotypes of out-groups, and the illusion of unanimity (Janis, 1982).
Two related phenomena are *group polarisation* and *risky shift*. The characteristic of group polarisation is that the predominant view in a group discussion is intensified during the session. The risky shift concerns the fact that a group tends to make riskier decisions, compared to how risky the decisions of each individual, independently of each other, would make (Bazerman, 1998; Miner, 1992).

The *escalating commitment* phenomenon is not a pure GDM problem. It can also occur with individual decision-makers (Miner, 1992) and must also be dealt with from an organisational point of view (Staw, 1997). An escalation dilemma occurs when things have gone wrong, but corrective actions can make things worse. A decision must then be made. Should the person carry on in the current situation or start anew? There is a tendency that people and organisations get stuck in a losing course of action, the escalation of commitment (Staw, 1997). This causes a group to invest more and more in the initial decision in order to justify a bad decision (Miner, 1992). This phenomenon can, for example, be one of the causes behind the problem discussed by Lyytinen and Robey (1999). They claim “many ISD [information systems development] organizations appear unable or unwilling to adjust their practices even when they fail to produce beneficial results” (Lyytinen & Robey, 1999, p 87). These organisations may have invested too much, such as knowledge, prestige, and money, in current practices, which makes it hard to change a practice even if it is obvious to an observer that it ought to be changed.

All together, these aspects that a group perspective adds to decision-making increase complexity, and need to be taken into account when supporting RE decision-makers, since they frequently carry out RE decision-making activities with other persons. Thus, group decision-making theories can inform developers of REDSS that also the social view of RE decision-making should be considered and facilitated.

However, since both individuals and groups are affected by the context in which they work, theories from the discipline of organisational decision-making can be a complement, which can deepen the understanding of decision-making.

### 3.3.6 Organisational decision-making

Organisational decision-making (ODM) has its roots in the work of Herbert Simon “Administrative behavior” from 1945, and his work with James March called “Organizations” published in 1958. March also wrote an important book together with Richard Cyert called “A behavioral theory of the firm” first published 1963 (Lipshitz et al., 2001).

ODM focuses on aspects characterising and affecting decision-making in an organisational context. Shapira (1997) describes five characteristics of ODM, which presumably can be used to describe a certain decision context:

- *Ambiguity* often has to be dealt with by decision-makers in an organisation, such as information ambiguity, preferences, and interpretation of the history of decisions.
- **Longitudinal context** concerns the context decision-makers act in, because decisions in organisations are part of an ongoing process.
- **Incentives**, and penalties, affect the decision-maker.
- **Repeated decisions** are often made by decision-makers. Especially persons in middle management make repeated decisions on similar issues, e.g., a loan officer makes repeated decisions on new loans.
- **Conflicts** are part of ODM, since organisations can be seen as political systems in which decision-makers act. Therefore, power considerations and agenda setting influence the decision-making.

These characteristics are similar to the list of factors characterising naturalistic decision-making (NDM, see section 3.3.4). Ambiguity and longitudinal context can be regarded as part of the uncertain environment. Incentives and penalties can increase the stakes for the individual decision-maker. Conflicts arise when there are multiple participants with different opinions, i.e. competing goals. The relations between behaviours and the factors of NDM and ODM are discussed further in this section.

The list of ODM characteristics shows that organisational decision-making is a complex process that needs to be studied with different approaches. Keen and Scott Morton (1978) present a classification of decision-making literature from an organisational point of view:

- The rational manager view
- The “satisfying” and process-oriented view
- The organisational procedures view
- The political view
- The individual differences perspective

The rational manager view overlaps with CDM and its normative decision theories described in section 3.3.4 on page 63. The satisfying, process-oriented view partly coincides with JDM. This view has, according to Keen and Scott Morton (1978), a descriptive focus. It also includes, for example, the work of Simon (1997) concerning bounded rationality, which is presented below. The organisational procedures view is concerned with the formal and informal structures of the organisation, organisational roles, procedures, and communication channels. The work of Cyert and March (1992), for example, is included by Keen and Scott Morton (1978) in this view. In the political view the bargaining process is important, and it also focuses on power and its influence on the decision. The last view, the individual difference perspective, includes aspects such as personal decision-making styles (Keen & Scott Morton, 1978), which is presented in section 3.2.2.

**The “satisfying” and process-oriented view**

Normative theories are based on the idea of rationality, especially objectively measurable rationality. However, as argued by Lindblom (1959) and Simon (1997), decision-making is not rational. Instead of behaving a rationally, Lindblom (1959) argues that the decision-maker searches incrementally for alternatives. These
alternatives do not differ so much from the existing situation. The focus is on marginal values. Evaluation and empirical analysis are intertwined, i.e. values and policies are chosen at the same time. The organisation is “muddling through” (Lindblom, 1959).

Simon (1997) has another way of arguing against the assumptions of rationality, stressing that rationality can have different meanings and limitations, which can be used to describe decision-making. He presents different meanings of rationality. A decision is:

- **Objectively rational** if it maximises the given values in a given situation, i.e. if it is in fact the correct behaviour.
- **Subjectively rational** if it, relative to the actual knowledge of the person, maximises the achievement.
- **Consciously rational** in relation to the degree of the consciousness concerning adjustment of means to ends.
- **Deliberately rational** in relation to the degree of deliberately bringing about adjustments of means to ends.
- **Organisationally rational** if it is directed to the goals of the organisation.
- **Personally rational** if it is oriented to the goals of the individual.

The first meaning of rationality, objectively rational, is the same as in normative decision-making theories. In those theories, the aim is to optimise the outcome of decision-making. Two other meanings of rationality, consciously rational and deliberately rational, resemble each other. Both types define rationality as a relation between the used resources and the outcome. The last two meanings of rationality, organisationally and personally rational, can be oriented towards the same specific goals. In such cases, decision-making can be relatively easy to conduct. In other cases, when the goals are competing, decision-making can be more difficult.

The different meanings of rationality raise questions that can be important to answer when developing an REDSS. For example, when evaluating alternatives, is organisational rationality most important or to what extent are personal goals taken into account? Is it more important to be objectively rational than to include the relation between potential outcome and the resources that have to be spent?

The second meaning of rationality, subjectively rational, is, in some senses, related to another concept invented by Simon (1997) called bounded rationality. Bounded rationality describes decision-making within constraining conditions:

- Incompleteness of knowledge and lack of information
- Difficulties of anticipation
- Scope of behaviour possibilities

It is the first of these conditions, *incompleteness of knowledge and lack of information* that is especially similar to subjective rationality. However, the other two can also be regarded as part of a decision-makers actual knowledge. *Difficulties of anticipation*
concern the difficulty of anticipating the actual consequences and the experiences of a consequence (Simon, 1997), which can be due to several reasons. One is that the consequences might be impossible to forecast. Another reason can be that the decision-maker’s knowledge is insufficient to enable prediction to an acceptable level of probability. A third possible explanation can be that the relationships between cause and outcome are too complex for human beings to manage. A similar discussion could be carried out for the condition *scope of behaviour possibilities*. Simon (1997) explains that this condition is the difficulty for a decision-maker to imagine all the possible ways of acting.

The satisfying, process-oriented view of ODM describes how decision-making in organisations is actually carried out and what goals are used. RE decision-making is always carried out in an organisation. Therefore, it is relevant to include decision-making theories from the satisfying, process-oriented view. Such theories can be used to extend the understanding of how RE decision-making is accomplished in an organisational context. They can also be used to elaborate the actual goals of RE decision-making.

Descriptive theories and models within this view may form a basis for developing solutions to the issue of reducing existing problems in decision-making. It can also be advantageous to include other views of ODM as a foundation for solutions, for instance, the organisational procedures view.

**The organisational procedures view**

Two theories are presented in this section. First, a description of Cyert’s and March’s (1992) theory of decision-making within business organisations, concerning how firms make economic decisions. Second, an explanation of the garbage can model by Cohen et al. (1972).

The theory that Cyert and March (1992, p 162) developed is primarily focused on “large, multiproduct firms operating under uncertainty in an imperfect market”. Their theory on organisational decision-making consists of two parts.

The first part is a framework consisting of a set of variables affecting organisational decision-making:

- Goals
- Expectations
- Choice

The second part concerns key relations among system variables:

- Quasi resolution of conflict
- Uncertainty avoidance
- Problemistic search
- Organisational learning
An example of a variable affecting organisational goals is past performance of other comparable organisations. The variables that have an effect on organisational expectations are either affecting the inference drawing process or the process in which information is made available to the organisation. One aspect of organisational choice that can be affected is standard decision rules, which, for example, can be affected by the variable called past experience of the organisation. The second part of the theory is, according to Cyert and March (1992), the heart of the theory. The first relation, quasi resolution of conflict, assumes that an organisation is a coalition of people having different goals, which forces the organisation to handle conflicts and consider latent conflict of goals. These conflicts are “resolved” in different ways. The second relation, uncertainty avoidance, states that organisations try to avoid uncertainties. Such an avoiding strategy is “avoiding planning where plans depend on predictions of uncertain future events and by emphasizing planning where the plans can be made self-confirming through some control advice” (Cyert & March, 1992, p 167). Problemistic search is a search directed towards finding a solution (but not necessarily an optimal solution), i.e. it is goal-driven. The last relation, organisational learning, includes three phases of organisational adaptation: adaptation of goals, adaptation of attention rules, and adaptation in search rules. Cyert and March (1992) stress that all these concepts are fundamental to understanding the decision-making process of a business organisation.

The relations in the theory of Cyert and March (1992) can be considered as activities carried out in organisations. These activities can be regarded as the result of factors that characterises NDM and ODM. The NDM factors (see section 3.3.4 on page 66) are characteristics of decision-making in any natural unspecified context. The ODM factors are characteristics of decision-making in the specific context of an organisation. NDM factors can then be assumed to be related to and have an effect on the ODM factors, which in turn can be assumed to cause certain behaviours in an organisation. These relations can be seen in Figure 21. The figure does not include all the factors defined by Shapira (1997). The excluded factor ‘incentives’ is not supposed to not be affected by the NDM factors in the same way as the other ODM factors. The figure is not intended to be a complete chart of the relationships between factors and behaviour, but rather an example of a way of exploring causes and effects. All behaviours are presumed to have several more causes and each factor is assumed to affect in several more ways.
One cause of the behaviour ‘trying to avoid uncertainty’ can be the ambiguity decision-makers in an organisation have to deal with. The ambiguities can be the effects of ill-structured problems, uncertain and dynamic environments, and shifting, ill-defined, or competing goals. The need of ‘trying to solve conflicts’ has, of course, its roots in the conflicts that are claimed to be a characteristic of ODM. In order to arise, conflicts need multiple players with competing goals and values. The longitudinal context of ODM and repeated decisions may raise a need to ‘learn within the organisation’, so that the decision-making competence increases. Since the ‘problemistic search’ is stated to be goal-driven, the NDM factor, organisational goals and norms, may direct the search. High stakes and time stress may also influence how the search for a solution is carried out.

This way of reasoning can, for example, be used when supporting certain behaviour in a decision situation. If organisational learning is meant to be improved in order to enhance the possibility of making more effective decisions in the future, then perhaps the information gained from action and feedback loops should be taken into account.

Another theory within the organisational procedures view is the “garbage can”, which was introduced by Cohen et al. (1972). The garbage can is a model that describes decision processes in organisations. This model shows a process in which problems are not solved well, but nevertheless enables choices to be made and problems to be solved, in spite of the conditions being ambiguous and poorly understood. According to the garbage can model, a decision is an “outcome or interpretation of relatively independent streams within an organization” (Cohen et al., 1972, p 2-3). Four streams are identified: problems, solutions, participants, and choice opportunities. A choice opportunity is when an organisation is expected to
make a decision. It can be viewed as a garbage can where problems and solutions are dumped by participants. Decision situations in organisations can be described in terms of organised anarchy. They are characterised by a) problematic preferences, i.e. the organisation works under inconsistent and ill-defined preferences, b) unclear technology, i.e. the processes of the organisation are not fully understood, and c) fluid participation, i.e. the participants come and go and each participant can provide a certain amount of effort which differs over time (Cohen et al., 1972).

The organisational procedures view is concerned with the formal and informal structures of the organisation, organisational roles, procedures, and communication channels in relation to decision-making. This is also applicable to RE decision-making and can provide additional comprehension of RE decision-making in an organisational context. Such understanding is important in order to positively influence the factors that affect the behaviour of RE decision-makers.

A third perspective of organisational decision-making is the political view, where an important aspect is power.

**The political view**

Organisational decision-making can be regarded as a political process (Klein & Methlie, 1990), where political models of decision-making can be developed (Browne, 1993). A central concern is power, which can be seen as the potential to influence people in how they act and think. According to Pfeffer (1992), power is important both in the activities preceding a decision and in the implementation of a decision. The amount of power involved in a decision process depends on the characteristics of a decision. Power is used to a greater extent in situations such as interdepartmental coordination, at the top management level, in functional areas such as marketing, as well as in decisions concerning e.g., reorganisations. Power is used to a lesser extent in situations such as work appraisals, at lower management levels, in functional areas such as production, as well as in decisions concerning rules and procedures (Pfeffer, 1992). Thus, it is advantageous to state the characteristics of a certain decision. Depending on the decision characteristic, it is possible to decide whether to use political decision models or not when developing decision support.

Harrison (1999) and Browne (1993) describe the characteristics of political decision-making models and the political perspective of decision-making. The focuses, views, and concerns are:

- Behaviour of individuals in organisations
- Compromise and bargaining strategies of decision-making
- Finding alternatives that are acceptable to all stakeholders
- Alternatives that differ from existing policies, which often imply a small number of options and a limited number of consequences
- Continual redefinition of the problem
- There is no right choice, only different ways of tackling the problem
- Short-term rather than long-term results
- Information is a resource that can give power depending on how it is used
• An organisation is a coalition of individuals with goals and ambitions, which forces the decision-making process to take this into consideration
• To what extent and in what ways the decision-making group is influenced by external stakeholders
• The power of individuals and roles within the organisation

The RE decision processes can probably also be regarded as political processes, like other organisational decision processes. It would be interesting to investigate the political perspective of RE decision-making and find out in what way such aspects influence the outcome of RE decision-making. It is perhaps possible to change or augment political aspects so that the benefits of high quality RE decision-making increase.

In summary, ODM concerns decision-making from an organisational point of view, where individuals and groups are natural components. Its purpose is, primarily, to describe what “actually happens”:

• Actual types of goals
• The actual activities and behaviours
• Factors affecting decisions

In order to effectively support RE decision-making, there is a need to understand the reality in which RE decision-makers act. Otherwise, there is a risk of supporting something that does not need support, or trying to support in an appropriate way.

### 3.4 Chapter summary and reflections

In this chapter, we present an overview of concepts and theories of decision-making. The term decision is viewed as both decision matter and decision outcome. Decision-making concerns the activities that a decision-maker carries out during a decision process. Decision-makers have their own psychological types and decision styles that influence their preferences in decision-making. Decision-makers can be classified as either an individual decision-maker or multiple decision-makers. An individual decision-maker can either be a person or a computer, while multiple decision-makers can be classified as team, group, or organisation. There are many decision-making theories, each one belonging to a certain research tradition. Problem solving theories are not viewed as decision-making theories, however closely related to decision-making. Related are also generic models of decision processes, in which the decision-making activities take place. Decision-making theories can be divided into three categories: individual, behavioural decision-making, group decision-making, and organisational decision-making. Individual, behavioural decision-making has the subcategories normative, descriptive, and prescriptive decision theories. Normative decision theories state how a person should choose an alternative in order to make an optimal decision in a quantitative sense. While descriptive decision theories describe how a decision-maker actually makes decisions, prescriptive decision theories propose how a person should act to make better decisions.
We argue that it is important to understand the decision situation from the decision-makers’ perspective in order to provide effective and efficient decision support. Requirements engineering (RE) can be regarded a decision process as previously elaborated (see section 2.5). Therefore, the general definitions of decision-making can be inherited by RE and theories and models of general decision-making can be transferred to RE decision-making. Such theories and models have the potential of providing us:

- A deepened understanding of what needs to be investigated concerning RE decision-making
- Explanations of empirical findings of RE decision-making
- Predictions of consequences of identified RE decision-making aspects and conditions
- Ideas for how to enhance RE decision-making
- Inspiration for how to offer RE decision support
- Rationale for REDSS characteristics and features

Hence, there are several benefits from using general theories of decision-making in research in RE. Yet, knowledge of decision-making in general is not enough to achieve a cohesive body of knowledge of RE decision-making. We also need domain-specific knowledge, which can be obtained by combining general “theoretical” RE with general decision-making theory and then making inferences for RE decision-making. However, this is not enough. We also need empirical foundations for developing a body of knowledge concerning RE decision-making.

An understanding of actual RE decision situations underpins the development of characteristics and guiding principles for how REDSS that embraces the whole of the RE decision processes should be constituted. To obtain ideas about how decision-making can be supported, we can use of the field of decision support systems.
4 Decision Support Systems

In this chapter, a brief introduction of decision support systems (DSS) is provided. To be able to support decision-makers, we not only need to understand decision-making, but also the potential support that can be afforded by a DSS. In this chapter, the potential support is described through definitions and characteristics of DSS, the different types of DSS and its components, ways of supporting decision-making and the benefits and limitations of DSS.

4.1 Definitions

There is no consensus concerning what a DSS is and how it should be defined. Some definitions focus on what a DSS does, and others focus on how to accomplish the DSS’s objectives. Definitions that are categorised as having a “what focus” consist mostly of concepts such as: the purpose of DSS, the people using them, and the type of problem that can be supported. An example of a definition that has a “what focus” is Keen and Scott Morton’s (1978, p 1): “Decision support implies the use of computers to: (1) Assist managers in their decision processes in semi-structured tasks. (2) Support, rather than replace, managerial judgment. (3) Improve the effectiveness of decision making rather than its efficiency.”

The definitions categorised as having a “how focus” consist of concepts such as system components and development process. Bonczek et al. (1981, p 69), for example, have a distinct “how focused” definition. They define DSS as having “…three principal components: a language system (LS), a knowledge system (KS), and a problem-processing system (PPS)”.

There are definitions that include both “what concepts” and how concepts”. One example is Turban’s (1990, p 109) definition: “A DSS is an interactive, flexible, and adaptable CBIS [Computer-Based Information System] that utilizes decision rules, models, and model bases coupled with a comprehensive database and the decision maker’s own insights, leading to specific, implementable decisions in solving problems that would not be amenable to management science optimisation models per se. Thus, a DSS supports complex decision making and increases its effectiveness.”

It is surprising that many DSS definitions contain system components (see e.g. Holsapple & Whinston, 1996; Sprague & Watson, 1979; Power, 2002), because components may differ between systems. In our view, the most important parts of a DSS definition are system objectives and problem type. These parts indicate what we are aiming at, i.e. to support decision-makers so they can make more effective decisions when dealing with semi-structured and unstructured problems. The others can differ over time and between systems, hence definitions containing such parts may be out of date. Therefore, the working definition of this thesis reads: A DSS is a
computer-based information system that supports either a single decision-maker or a group of decision-makers when dealing with unstructured or semi-structured problems in order to make more effective decisions. The DSS supports one or more decision-making activities carried out in a decision process.

When we convert this general definition of a DSS to a requirements engineering decision support system (REDSS) it reads: An REDSS is a computer-based information system that supports either a single RE decision-maker or a group of RE decision-makers when dealing with unstructured or semi-structured RE problems in order to make more effective decisions. The REDSS supports one or more RE decision-making activities carried out in an RE decision process.

A definition gives us a starting point for painting a picture of what a DSS can be. The next step is to outline the characteristics a DSS can have.

### 4.2 Characteristics

Since there is no consensus concerning what DSS is, there is no consensus on standard characteristics (Turban et al., 2007). Instead, there are a number of characteristics, where some are more commonly agreed on, and others more rarely mentioned in the literature. All the characteristics are not included in every DSS (Mallach, 1994).

A DSS is a computer-based, interactive information system, i.e. it inherits the qualities about information systems in general. The term interactive implies that there is an exchange between the system and the user. A DSS primarily supports managerial activities at various levels. The purpose of a DSS is focused on improving the effectiveness of the decision-making process, rather than its efficiency. The effectiveness of decision-making concerns timeliness, accuracy, and quality, while efficiency is the cost of making the decision, e.g., cost of the decision-maker's working hours (Alter, 1980; Bidgoli, 1989; Keen & Scott Morton, 1978; Mallach 1994; Marakas, 1999; Sprague, 1989; Turban et al., 2007).

DSS provides support for decision-makers when they deal with semi-structured and unstructured problems. Support is provided in all four phases of the decision-making process, i.e. intelligence, design, choice, and implementation. Thus, focus can be both on decision-making as well as implementation of decisions. A DSS may provide support for both interdependent and multiple independent decisions (Bidgoli, 1989; Mallach, 1994; Marakas, 1999; Turban et al., 2007).

Decision-makers use a DSS actively, which means that the user initiates every instance of use, and should be in complete control of the decision process. Furthermore, the DSS should support, not replace the decision-maker. A DSS can support learning, so that the decision-maker can be trained to perform better in future decision situations (Alter, 1980; Bidgoli, 1989; Keen & Scott Morton, 1978; Mallach, 1994; Marakas, 1999; Turban, 1990; Turban & Aronson, 1998).
Decision-makers should be able to confront changing conditions. Therefore, a DSS has to be *adaptive and flexible* in order to meet the needs of decision-makers. There is an emphasis on ad hoc utilisation. A DSS should be *easy to use*. Support is provided to *individuals and groups*, and a DSS can be tailored to support different *decision-making processes and decision styles*, in order to fit the individual decision-maker (Alter, 1980; Bidgoli, 1989; Marakas, 1999; Sprague, 1989; Turban et al., 2007).

*End-users* should be able to construct and modify a simple DSS themselves. In order to support the judgment of decision-makers, *analytical techniques* should be provided by the DSS. A DSS also incorporates *models* that enable experimenting with shifting conditions and *data* from a variety of sources, formats and types (Bidgoli, 1989; Keen & Scott Morton, 1978; Mallach 1994; Marakas, 1999; Sprague, 1989; Turban et al., 2007).

These general characteristics of a DSS can be inherited by an REDSS. They can inspire us in the creation of empirically based domain-specific characteristics of an REDSS. As mentioned above, not all of these characteristics are present in every DSS. The same is true for our REDSS characteristics. However, they show the possible scope of an REDSS.

A reason why not all characteristics are present in every DSS is that there are several different types of DSS. Each type focuses on supporting decision-making in a certain way. Depending on the type of DSS some characteristics are more present than others, and different types of benefits can be gained.

### 4.3 Types of DSS

There are different types of DSS and one way to categorise decision support systems is provided by Power (2002). He introduces a framework, in which the term ‘driven’ is used, that points at the dominant functionality of the DSS. Power’s (2002) categories are:

- Data-driven DSS
- Model-driven DSS
- Knowledge-driven DSS
- Document-driven DSS
- Communication-driven and group DSS.

*Data-driven DSS* provide access to large amounts of data and support analysis. They enable display and manipulation of data sets. Data-driven DSS can be divided into the subcategories: data warehouses, on-line analytical processing (OLAP) systems, executive information systems (EIS), and spatial DSS (Power, 2002; Turban et al., 2007). A data warehouse is defined as a “subject-oriented, integrated, time-variant, non-volatile collection of data in support of management’s decisions” (Inmon & Hackathorn, 1994, p 2). It is concerned with the major subjects of an organisation, and provides a base for integration of a separate system. The data can have a historical perspective, and the non-volatility characteristic means that “data is loaded into the
warehouse and is accessed there, but once the snapshot of data is made, the data in
the warehouse does not change” (Inmon & Hackathorn, 1994, p 10). Through data
mining a decision-maker can obtain “answers” from a data warehouse. Data mining
is defined as “the set of activities used to find new, hidden, or unexpected patterns in
data” (Marakas, 1999, p 356). On-line analytical processing (OLAP) is software
technology that carries out multidimensional analysis of data (Marakas, 1999). An
EIS is a “computer-based system intended to facilitate and support the information
and decision-making needs of senior executives by providing easy access to both
internal and external information relevant to meeting the stated goals of the
organization” (Marakas, 1999, p 185). Spatial DSS are described by Seffino et al.
(1999, p 105) as “decision support systems where the spatial properties of the data to
be analysed play a major role in decision making”.

Model-driven DSS mainly provide support through models, e.g., financial or
optimisation models (Power, 2002; Turban et al., 2007). According to Shim et al.
(2002), a model-based decision support includes three stages: a) formulation, i.e.
generation of an acceptable model, b) solution, i.e. the algorithmic solution of the
model, and c) analysis, i.e. the what-if analysis and interpretation of the model
solutions. Model-driven DSS can be compared with spreadsheet-oriented DSS
(Holsapple & Whinston, 1996). Spreadsheets can be used to create models and do
what-if analysis, and are often used in end-user developed DSS (Turban & Aronson,
1998).

Knowledge-driven DSS consist of knowledge, understanding of problems, and
problem solving “skills” within a specific domain (Power, 2002; Turban et al., 2007).
Knowledge-driven DSS are related to, e.g., rule-based DSS and intelligent DSS
(Power, 2002). Techniques from artificial intelligence (AI) and expert systems are
used in knowledge-based DSS. With the help of these techniques an intelligent DSS
behaves in a better (more “intelligent”) manner (Turban & Aronson, 1998). A rule-
based system is “a system in which knowledge is represented completely in terms of
rules (for example, a system based on production rules)” (Turban & Aronson, 1998, p
867).

Document-driven DSS focus on gathering, retrieving, classifying, and managing
unstructured documents, and where a search engine can be a useful tool. Such a
system can deal with, for example, policies, procedures, and product specifications
(Power, 2002; Turban et al., 2007). There are materials that decisions can be based on,
which are not ordinary data and therefore cannot be put in a database, e.g., letters
from customers, written reports, and news items. This information also needs to be
handled in a DSS, and therefore information retrieval is important (Federowicz,
1989). Document-driven DSS can be compared to text-oriented DSS (Holsapple &
Whinston, 1996), which keep track of textually represented information. Hypertext is
a technique that can be used in text-based DSS (Holsapple & Whinston, 1996;
Marakas, 1999).

Communication-driven and group DSS, where communication-driven DSS focus on
supporting collaboration, communication, and coordination, while group DSS
(GDSS) focus on supporting groups of decision-makers in analysing problem situations and performing group decision-making tasks (Power, 2002; Turban et al., 2007). Examples of tools that support communication between decision-makers include web conferencing, interactive whiteboards, screen sharing, and online workspaces (Turban et al., 2007). Electronic Brainstorming that generates stimulating questions to the assembled participants (Power, 2002) is an example of a GDSS.

Several types can be of use for an REDSS. For example, requirements and their related information can be stored in a database, which points out a data-driven DSS. Moreover, several different types of documents are produced in RE and even more are used in it. This indicates that a document-driven DSS can be useful. Furthermore, there are several stakeholders in RE decisions, which imply that a communication-driven or a group DSS can be beneficial. Thus, there are multiple ways of supporting decision-making.

### 4.4 Supporting decision-making

There are various approaches to support decision-making using a DSS, and there are two different main kinds of support for decision-makers. The first is to support the decision performance in a specific decision situation, and the other is to train decision-makers so they can perform better in the future (Zachary & Ryder, 1997). An example of supporting decision performance is presented by Benbasat and Lim (2000), which aims to reduce availability bias (see section 3.3.4) in group judgement. They found that an electronic brainstorming tool increases the number of generated ideas and decreases the availability bias. An example of decision support through training decision-makers is provided by McGrath and More (2001), who describe the Greta system. This system is primarily a pedagogical aid, based on a power-political model of organisational decision-making. In the Greta system, the users are faced with the task of successfully implementing a customer support system, where each simulation cycle begins with a number of tactics from which the user chooses.

Another way of presenting approaches for decision support is provided by Silver (1991), who claims that substantive decision support addresses one or more of the following parts:

- Decision-making process
- Decision-making needs
- Decision-making environment

Each of these three can be viewed as supporting either decision performance or decision training. However, it can be assumed that Silver (1991) considers them to be focused on decision performance.

When the decision-making process is in focus, the support of a DSS is directed to reducing effects of human decision-making weaknesses or cognitive limitations in general (Silver, 1991). Holsapple and Whinston (1996) provide two ways of supporting this: a) facilitating or extending the user’s ability to process information,
i.e. acquiring, transforming, and exploring information, and b) stimulating the
perception, imagination, and creative insights of the decision-maker. An example of
the first way is the exploratory cognitive DSS for strategic decision-making
developed by Chen and Lee (2003). The purpose of this DSS is to reduce particular
cognitive biases in decision-making, such as availability and reason by analogy, with
the help of a case memory, cognitive mapping, and scenario building. An example of
the second way is the creativity enhancing decision-making support system
developed by Forgionne and Newman (2007). They describe that the system is used
by the decision-maker to organise the problem knowledge, structure ideas and
concepts into problem elements and relationships, and simulate conceptual problem
solutions.

If the prime concern is to support the decision-making needs, then the DSS is aimed to
support the identified needs of a decision-maker in a certain decision situation. These
needs are identified by the decision-maker or an analyst. Examples of needs can be
lack of necessary information or problems in generating alternatives (Silver, 1991).
Four ways of support in this approach are: a) a decision-making opportunity and
challenge is alerted to the decision-maker, b) recognising problems that need to be
solved in a decision-making process, c) problem solving, and d) the decision-maker
is offered advice, expectations, evaluations, facts, analyses, and designs (Holsapple
& Whinston, 1996). Two examples of DSS developed to support in a certain decision
situation are the Performance-Net (Ioannou & Mavri, 2007) and a decision support
system for housing evaluation (Natividade-Jesus et al., 2007). The purpose of the
Performance-Net is to support the management of a bank in establishing measurable
branch goals as well as evaluating performance and planning for new branch
locations (Ioannou & Mavri, 2007). The purpose of the multi-criteria DSS for housing
evaluation is to assist persons in the housing market make better-founded decisions
(Natividade-Jesus et al., 2007).

A DSS that considers the decision-making environment goes beyond the individual
decision-maker and also includes implications of the context. This can embrace, for
instance, organisational or group settings. A way of doing this is to coordinate and
facilitate relations between multiple decision-makers (Holsapple & Whinston, 1996).
Two examples of this are decision conferencing and computer-mediated
communication. Mustajoki et al. (2007) describe an interactive computer support for
decision conferencing in off-site nuclear emergency management. A decision
conference is a moderated decision analysis workshop. Pisarra and Jesuino (2007)
present idea generation through computer-mediated communication.

For an RE decision-maker, it can be beneficial to be supported in all these ways. It
can be valuable to reduce the weaknesses of human decision-making, for example,
by reducing the cognitive load of RE decision-makers. Furthermore, it can be useful
to have tailored decision support for specific RE decision-making tasks, such as
prioritising requirements. In addition, it can be helpful to include contextual support,
for instance, facilitate communication with stakeholders of RE decisions. Thus,
depending on the current type of support various benefits can be gained.
4.5 Benefits and limitations

There are many benefits from using a DSS, such as creating advantages over competing organisations (Marakas, 2003). However, every DSS does not provide all possible benefits. DSS also have limitations, such as that the “knowledge” and “skills” of a DSS are constrained and it cannot “perform” creativity and imagination (Marakas, 2003). The benefit categories are (Alter, 1980; Power, 2002):

- Improve individual productivity
- Improve decision quality and problem solving
- Facilitate interpersonal communication
- Improve decision-making skills
- Increase organisational control

By improving individual productivity, it is possible to save time associated with tasks connected to decision-making (Alter, 1980; Keen, 1989; Marakas, 2003; Turban et al., 2007). This means the tasks should be accomplished in less time, be carried out more thoroughly in the same amount of time, or more appropriate tasks should be executed with less effort. This can be done by increasing the number of alternatives examined. For example, solutions imagined by the user, can be tested or simulated. With a proper DSS tool it should also be possible to make better use of data resources (Keen, 1989). The ability of decision-makers to process information and knowledge can be extended (Marakas, 2003).

In order to improve decision quality and support the overall problem solving, the ability of decision-makers to tackle large-scale, time-consuming, complex problems can be extended (Marakas, 2003). The DSS makes it possible to give fast responses to unexpected situations and to do ad hoc analyses. In addition, the quality of problem solving can also be enhanced. Through a better understanding of the business, e.g., improved abilities to see relationships between variables, and through increased depth and sophistication of analysis, improved decisions can be made (Keen, 1989). This can also be done by revealing new approaches of dealing with the problem (Marakas, 2003).

A DSS can facilitate interpersonal communication (Alter, 1980; Keen, 1989; Turban et al., 2007), by providing communication support in at least two ways. It provides decision-makers with tools for persuasion and facilitates communication across organisational boundaries (Alter, 1980). The individual decision-maker can obtain substantiated arguments, which can be particularly useful when implementing decisions. As claimed by Marakas (2003), a DSS provides enhanced possibilities for generating new evidence in confirming existing assumptions and reliability of outcome. It is not clear what Marakas (2003) means by reliability of outcome, but it can be interpreted as enhancing the possibilities of evaluating possible consequences, for example, with the help of simulation. Alter (1980) claims that communication between organisational units can be made through standardising the mechanics and vocabulary of negotiation and by providing a common conceptual basis. Groupware
provides additional communication paths, which may improve the communication (Mallach, 1994) and obtain more effective teamwork (Keen, 1989).

A DSS can improve decision-making skills (Power, 2002). A DSS can promote both organisational and individual learning, for instance, by making it possible for decision-makers to obtain a better understanding of the business. An example is a decision-making team in a company that used a strategic planning system. The decision-makers claimed that through using this system they obtained a better understanding of the strengths of the business, the constraints under which it operates, and what manoeuvring room was available (Alter, 1980). Marakas (2003) confirms this view and suggests that a DSS can lead to new insights and learning, which can encourage decision-makers to explore.

A DSS can increase organisational control (Alter, 1980; Power, 2002). Organisational norms and requirements can constrain the individual decision-maker and ensure consistency across organisational units which can be made clear to the decision-makers (Mallach, 1994). Summary data can be provided by a data-driven DSS and can be used by managers for organisational control purposes (Power, 2002).

There are, as described, a number of benefits. It is also important to measure the success of an implemented DSS, which can be done using some DSS success measures, summarised by Hung et al. (2007). There are two main categories of DSS success measures; process-oriented and outcome-oriented. Process-oriented measures include frequency or length of system usage. Efficiency is an example generally measured by decision speed or the number of alternatives that are considered. Outcome-oriented measured include decision performance and user satisfaction. For example, effectiveness is measures by decision outcome, e.g., user satisfaction and quality or accuracy of decision (Hung et al., 2007). Some of these success measures are probably not unique for DSS, but are to some extent the same for information systems in general.

There are also limitations to DSS use, not the least of which is that a good DSS cannot compensate for a bad decision-maker (Marakas, 2003). Marakas (2003) and Power (2002) list some limitations:

- A DSS cannot have human decision-making abilities, such as creativity, imagination, or intuition.
- A DSS is limited by its stored knowledge, data and models as well as by the operating computer system.
- The user interfaces are not sophisticated enough for full interaction between the user and the system in natural language.
- It is difficult to design a general DSS that is applicable in multiple contexts, but instead they often have a narrow scope of application.
- Often, a DSS needs to be integrated into decision processes.
- A DSS can only be supportive if a decision-maker chooses to use the system and integrates the analyses into ‘off line’ thinking and analysis.
DSS is a type of behavioural engineering, and many managers refuse to accept such intrusions.

There can also be a problem, for instance, with trust or responsibility problems. The decision-maker must trust the DSS in order to really use it in decision-making. In addition, there can be disagreements about who is responsible for the decision. Is it always the decision-maker or can the persons behind the DSS be blamed for a bad decision?

All these generic benefits of DSS are also attractive in RE. Thus, REDSS can be a way to gain these benefits. Unfortunately, the generic limitations of DSS can also be transferred to REDSS and thus need to be considered when developing and implementing such a system.

4.6 Chapter summary and reflections

In this chapter, we introduce the fundamentals of decision support systems (DSS). A DSS is a computer-based information system that supports either a single decision-maker or a group of decision-makers when dealing with unstructured or semi-structured problems in making more effective decisions. The DSS supports one or more activities in a decision process. A DSS can be data-driven, model-driven, knowledge-driven, document-driven, communication-driven or a group DSS depending on the dominant functionality. Furthermore, a DSS can either support the decision-maker in an on-going decision situation or it can prepare the decision-maker to perform better in the future through decision training. The benefits of using a DSS are that it can improve individual productivity, improve decision quality and problem solving, as well as facilitate interpersonal communication. It can also improve decision-making skills and increase organisational control.

A major issue in this thesis is to suggest a visionary requirements engineering DSS (REDSS). Therefore, the field of DSS can provide us with information that can be utilised in this work. The DSS domain has the potential of affording, for example:

- Foundations for elaborating how an REDSS can be characterised and what it can consist of
- Suggestions for how decision support can be provided
- Ideas for how to evaluate REDSS success

This concludes the presentation of the theoretical foundations of this thesis. Next, we describe and discuss our methodological considerations and our research process.
5 Research process

In this chapter, we present methodological considerations, the research process, and reflections on the research process. The nature of the research problems motivates a qualitative research approach as well as a design science approach, which has guided the design of the research process. The research process consists of four stages; a) a literature study that resulted in the decision situation framework, b) a case study that is the foundation for characterising the decision situation of requirements engineering (RE) decision-makers, and c) a synthesis of the empirical results and the theoretical work that underpin the desirable characteristics of RE decision support systems (REDSS) and the guiding principles that direct further efforts concerning how to find a solution which can fulfil the characteristic.

5.1 Methodological considerations

Our research project has been influenced by the theoretical traditions constructivism and pragmatism. For example, they have had an effect on the choice of approaches; a qualitative research approach and a design science approach.

5.1.1 Influences

The theoretical traditions that have primarily influenced us are constructivism and pragmatism. The premise of constructivism is that reality is socially constructed. Individuals interpret and construct their reality and “the world of human perception is not real in an absolute sense, as the sun is real, but it is ‘made up’ and shaped by cultural and linguistic constructs”. A constructivist studies the multiple realities constructed by individuals (Patton, 2002, p 96). This means that we view the experiences of persons and how they perceive their world as vitally important. As a consequence, empirical studies of humans in the topic (RE decision-making) were a natural choice. Empirical studies that are influenced by constructivism call for a qualitative research approach. Additionally, this influence also means that we do not view one person’s experience as more correct and true than another person’s experience.

To have a pragmatic view obliges us to focus upon consequent phenomena instead of antecedent phenomena, and upon possibilities for actions instead of precedents (Dewey, 1931). This means that the interpretations we make must make sense practically and that we have an interest in “not only for what ‘is’, but also for what ‘might be’ (Goldkuhl, 2004, p 13). As a consequence, our purpose is not only to describe decision-making in RE, but also suggest how RE decision-making can be supported. This means that a design science approach is appropriate.

Hence, constructivism and pragmatism have had an influence on the specification of research problems as well as the methodological decisions.
5.1.2 Approaches

The starting point of research is a problem or a hypothesis, sometimes vague and sometimes clear and explicit. The research is heading somewhere, which can be expressed in aims, objectives, and research questions. Since the route between the starting point and the goal is the research method, the researcher has to choose a suitable route, i.e. approach and methods, which leads to the goal. Hence, the guiding star for methodological considerations should be what we want to accomplish. The methodological decisions depend on which type of answer suits the question. Some questions need numerical answers and other questions need characterising descriptions (Patton, 2002; Repstad, 1993).

The essence of this research project is to a) understand how RE decision-makers experience their decision situation and b) suggest possible ways of supporting RE decision-making. When the nature of a research problem is to grasp the meaning or characteristics of the experiences of individuals a qualitative research approach is appropriate (Strauss & Corbin, 1998). Hence, qualitative data collection methods and qualitative data analysis techniques are the most suitable alternatives for the first part of our research problem. For the second part of the problem a design science approach is appropriate. Hevner et al. (2004, p 75) describe this as a problem solving paradigm that “seeks to extend the boundaries of human and organizational capabilities by creating new and innovative artifacts”.

Qualitative research approach

Patton (2002) proposes twelve themes that function as a strategic framework for qualitative research. These themes contain fundamental postulations and epistemological ideals. The first three themes of the framework are design strategies:

- Naturalistic inquiry
- Emergent design flexibility
- Purposeful sampling

Naturalistic inquiry denotes that the research should be carried out in a real-world setting and that the phenomenon of interest should not be manipulated. It also denotes that the appropriate interview technique is open-ended, conversation-like questions. Emergent design flexibility means openness to ambiguity and uncertainty in the research process and to avoid getting locked in a rigid design of the research process. Discoveries can show new paths in the process. Purposeful sampling suggests that since small samples cannot be generalised – still quite a lot can be learned from them – it is important to select the cases for study carefully. The cases should be information rich and enlightening (Patton, 2002).

The next four themes are strategies in a qualitative approach that concern data collection and fieldwork (Patton, 2002):

- Qualitative data
- Personal experience and engagement
• Empathetic neutrality
• Dynamic systems

Qualitative data should describe, and consist of someone’s experience expressed in citations, observations, and extracts from documents. The theme personal experience and engagement stands for the nature of the fieldwork. The researcher should get in direct and personal contact with the persons of interest and their natural environment. Empathetic neutrality and mindfulness concern the emotional and cognitive behaviour of the researcher. He or she should find the balance between getting too involved and staying too distant. If the researcher becomes too involved, judgement can be clouded and if the researcher stays too distant from the persons of interest a lack of understanding can occur. Dynamic systems denote that the qualitative researcher should pay regard to the changes and view the dynamics of persons and their situations as natural and expected. Thus, the researcher should not try to control the phenomenon of interest (Patton, 2002).

The last five themes provide guidance and directions concerning data analysis. The analysis strategies are as follows (Patton, 2002).

• Unique case orientation
• Inductive analysis and creative synthesis
• Holistic perspective
• Context sensitivity
• Voice, perspective, and reflexivity

Unique case orientation stands for being true to the unique individual cases and describe the unit of analysis, whether it is a person, an event, or a community. The case should be described holistically in depth and detail. Inductive analysis and creative synthesis mean that general patterns evolve from the findings in the cases. A theory emerges from the categories and relationships that are found in the data. The theme holistic perspective embraces the viewpoint that the phenomenon under study is too complex to be reduced to a few variables and a cause-effect relationship. Instead, the natural context must be present during the analysis in order to understand the complex interdependencies and dynamics. Context sensitivity stresses the importance of the natural setting, which, for example, includes temporal, physical, cultural, and historical aspects. This can be compared to laboratory experiments, which strive to generate context free findings that can be generalised. In qualitative research, transferability is often more desirable than generalisation. The last theme is voice, perspective and reflexivity. This means the researcher in qualitative methods is the instrument and is therefore part of the context of the findings. Consequently, the personal voice and the perspective of the researcher should be used and the voice and perspective should be explicit in writing about the findings. The researcher also has to “sharpen the instrument”, which means that the researcher has to be self-aware and reflexive. To be reflexive “is to undertake an ongoing examination of what I know and how I know it” (Patton, 2002, p 64).
These themes of the strategic framework for qualitative research have permeated the design of our research process. The case study took place in a natural setting and the context was not manipulated. The interview technique was open-ended questions and a conversational style was used. The research design emerged during the process and was not firmly planned from the beginning. There was only one case and the chosen interviewees were not many – but information rich. The data was in the interviewees’ own words, since all but two interviews were recorded and transcribed. We had personal and direct contact with the interviewees. The analysis was inductive and general patterns emerged from the empirical data. A holistic perspective was taken on the RE decision-makers’ decision situation and contextual information was included. Thus, the themes of the strategic framework for qualitative research were taken into account in the research process.

Design science

Design science is concerned with creation and evaluation of information technology (IT) artefacts in order to meet identified IT-related needs, e.g., business needs. Its goal is utility (Hevner et al., 2004). Hevner et al. (2004, p 81) claim that design science research “addresses important unsolved problems in unique or innovative ways or solved problems in more effective or efficient ways”. The creation of artefacts should be based on an existing knowledge base, e.g., theories, frameworks, methods, data analysis, and measures. This way rigor can be achieved.

There are four types of design science products, i.e. IT artefacts (March & Smith, 1995):

- Constructs
- Models
- Methods
- Instantiations

Constructs constitute a conceptualisation, i.e. a vocabulary of a domain, that can be used for describing the problems and specify their solutions. The constructs can be combined into models that can represent situations as problem and solution statements. From a design science viewpoint, a valuable model does not just describe how something is. Instead, it should be useful, for instance, when designing an information system. A third type of design science product is methods which is a set of steps that should be carried out in order to reach certain goals. It can, for example, be algorithms, descriptions of ‘best practices’, or guidelines. Instantiations are physical implementations based on the constructs, models, or methods, with the purpose of demonstrating their feasibility and effectiveness (March & Smith, 1995). IT artefacts can be represented in different ways, such as software, formal logic, and natural language descriptions (Hevner et al., 2004).

There are two activities in the research cycle of design science (March & Smith, 1995):

- Build
- Evaluate
These two activities can be compared with the two research activities of natural science and behavioural science; discover and justify (March & Smith, 1995; Hevner et al., 2004). First, you have to build an artefact (discover a phenomenon). Then, you can evaluate it (justify it). Building is “the process of constructing an artefact for a specific purpose” and evaluation is “the process of determining how well the artefact performs” (March & Smith, 1995, p 254). March and Smith mention that a difficulty of design science is that the performance of an artefact is context-dependent.

Due to time limitations, we focused on one part of the research cycle, i.e. the building activity. It has not been possible to carry out the full research cycle. The type of artefacts that we constructed is methods in natural language. We created guidance to development of REDSS.

Thus, the basis for our research process was a qualitative research approach as well as a design science approach.

5.2 The research process

In this section, we elaborate the research process. An overview is first provided, followed by a detailed description of each of the three stages.

5.2.1 An overview of the process

With the qualitative research approach as well as the design science approach in mind, the research process was designed and carried out. The research process consists of three stages as described in Figure 22.

![Figure 22, Overview of the research process]

To summarise the research process, three main activities were carried out, which are as follows:
1. The research process began with a literature analysis. Its purpose was to identify the key aspects related to a decision-maker in a decision situation as well as to obtain the theoretical foundations of human decision-making that are relevant for decision support. The literature analysis embraced a variety of decision-making theories and literature from the field of decision support systems. This stage of the research process resulted in a generic decision situation framework, presented in chapter 6. The decision situation framework was used in stage two of the research process.

2. The case study was carried out at a systems engineering company. Its purpose was to study RE decision-making in practice. The data collection techniques comprised open-ended interviews and a focus group session. The interviewees were requirements engineers and their related stakeholders. The decision situation framework was used as a means of structuring and guiding the data analysis. The results from the case study was a portrayal of the decision situation of RE decision-makers, more specifically in terms of a) decision matters, decision-making activities, and decision processes in RE (see section 7.1), b) information sources used by RE decision-makers (see section 7.3), and c) factors that affect the RE decision-maker (see chapter 8). The decision situation of RE decision-makers underpins the desirable characteristics of a visionary REDSS that were developed in stage three.

3. The third stage synthesised the empirical findings from the case study especially with work from the DSS field, but also from human-computer interaction and information visualisation. Its purpose was to identify desirable characteristics for a decision support system that in a holistic way support RE decision-making. This resulted in empirically based desirable high-level characteristics of a visionary REDSS and guiding principles, empirically as well as theoretically grounded. In addition, available techniques are presented together with the guiding principles as a range of potential means.

During stage two and three, complementary literature analyses were carried out, in order to deepen the understanding of the findings, identifying available techniques as well as obtaining arguments for suggestions.

In the following, a more thorough presentation of the different stages of the research process is provided.

5.2.2 Literature analysis – development of a decision situation framework

The analysis began with decision support system (DSS) literature in order to define the kind of knowledge needed when developing a DSS. Then literature from different research domains related to decision-making was included in the survey. This literature addresses how decisions can be defined and characterised, psychological types and decision styles, problem solving, and decision processes. Different types of individual behavioural decision-making theories were included:
normative, descriptive (both “traditional” and natural decision-making), as well as prescriptive. The literature analysis also covered the research on group decision-making and organisational decision-making.

The search for literature was first broad, in order to find and embrace all relevant aspects of decision-making from the decision-maker’s perspective. The analysis of literature was conducted in a systematic way. During the reading of papers and books, their contents were summarised in a document. These summaries were then grouped and the content groups (categories) were given denotations. The content groups were then structured in a hierarchical way. When no new content groups emerged, the search for more literature was more focused in order to deepen the understanding of each content group. The result from this step was synthesised into a generic framework that describes the different aspects of decision-making – the decision situation framework (see chapter 6) - where the decision-maker is placed in the centre.

The strength of the literature analysis is the scope. Since several different and related areas were covered it was possible to create a holistic view of decision-making from the decision-makers’ perspective. The weakness of the literature analysis is that it is somewhat shallow. There was a need to make a trade-off between breadth and depth, due to time limitations, and breadth was prioritised in this case, since we chose a holistic perspective. Having this perspective means we try to view decision-making as a whole, to be understood as a complex system that is greater than the sum of its parts (Patton, 2002).

5.2.3 Case study – investigating the decision situation of RE decision-makers

The aim of this part of the research project was to go beyond the generic framework by gaining more in-depth knowledge about the decision situation of RE decision-makers specifically. In order to gain this knowledge, case study was chosen as the research strategy. Case study is defined by Yin (1994, p 13) as an “empirical inquiry that investigates a contemporary phenomenon within a real-life context, especially when the boundaries between phenomenon and context are not clearly evident”. In our case, we focused on requirement engineers in a systems engineering company.

The approach is human-centred and focuses on the experiences of the requirements engineers and the actors in the systems engineering process, which are closely associated to them. Requirements engineers are regarded in this case as RE decision-makers. The purpose of the case study was to investigate the actual decision situations of RE decision-makers. We carried out the case study at a company that develops highly advanced systems.

A theoretical sampling approach was applied. Theoretical sampling means that the sampling is cumulative and that it is not predetermined from the beginning. Instead, the sampling evolves during the research process. Each sampled interview “builds from and adds to previous data collection and analysis” (Strauss & Corbin, 1998, p
The research was carried out in five stages: a) open-ended interviews with requirements engineers, b) a focus group, c) open-ended interviews with other actors in the systems engineering process that are related to requirement engineers, d) complementary interviews with both requirements engineers and other actors, and e) group interviews with IT consultants from another company concerning the potential transferability of the results.

The case

For confidentiality reasons, the description of the case is not thorough. The case study organisation was a company in Sweden, which has several highly technologically advanced software-intensive products, i.e. the products consist of hardware as well as software. The products comprise data-intensive embedded systems as well as user interfaces that mediate the critical interaction between the user and the product. If a product is ill-functioning or erroneously used, there can be disastrous consequences.

The products are sold to and used by large organisations of similar types. There are relatively few potential customers available on the market. Often, the customers have specific product requirements and therefore each product needs to be tailored. A typical development project for the company is long and expensive and only a few units of each product are delivered. A project in which a product is tailored to a customer’s needs involves between 3000 and 6000 requirements depending on the type of product. The requirements are of several types, and examples of crucial types are real-time and performance requirements. Such a project typically runs for 2-4 years and consists of about 25 000 man hours. When a new product is developed a typical project runs for 5-10 years and consists of about 250 000 man hours.

The case study was carried out during two years. In the beginning of the case study a waterfall development process was used in the company. Later on they start using a variant of Rational Unified Process (RUP). As seen in Figure 23, a project begins when a contract is signed with a customer. Sometimes, the product is to be integrated with the customer’s existing products, which results in several “folders” of customer requirements. At the system level the customer requirements are transformed into system requirements, which are then allocated to different subsystems. Subsequently, the requirements engineers create subsystem requirements based on the allocated system requirements. A subsystem consists of between 700 and 1300 requirements. Some functions, and thereby also requirements, are used by two or more projects at the same time. The subsystem requirements were often, but not always, stored in an RE tool. They used Serena RTM.

Figure 23 also shows the focus of concern in this research project, i.e. requirements engineers viewed as RE decision-makers, and the stakeholders that were interviewed.
Interviews I

In the first interview round, five requirements engineers were interviewed. The interviewees, who were suggested by our contact person, had worked between 4 and 25 years at the company and had been requirements engineers for between 1 and 10 years. Open-ended interviews were used, since “open-ended responses permit one to understand the world as seen by the respondents. The purpose of gathering responses to open-ended questions is to enable the researcher to understand and capture the points of view of other people without predetermining those points of view through prior selection of questionnaire categories” Patton (2002, p 21). The interview technique was inspired by analysis of information utilization (Gulliksen et al., 1997) and contextual inquiry (Holtzblatt & Jones, 1993).

Analysis of information utilisation is a technique that provides guidance about how to describe and analyse the way information entities are used in work situations. Work tasks are identified in terms of judgement and decision-making situations (Gulliksen et al., 1997). We used their advice on how to identify decisions in our interviews.

The purpose of contextual inquiry is to provide an understanding of current work practice and fundamental work concepts, which is to be used in system design to reach usability. Three principles underpin the contextual inquiry process: context, partnership, and focus. The first principle denotes that the most appropriate way to understand the work of other persons is to talk to them in their natural work environment. The second principle stands for creating an equal dialogue between the interviewer and the interviewee concerning a few areas. The third principle means that the interviewer should try to avoid getting caught in false or too narrow
presumptions of the interviewee’s work by actively trying to expand the focus area (Holtzblatt & Jones, 1993). All three principles were kept in mind during the planning and execution of the interviews. All the interviews were conducted at the workplace of the interviewees. As prescribed by Holtzblatt and Jones (1993) the questions were not standardised, and instead, an interview guide was used. This guide helps to ensure that the basic parts of the interviews remain the same and reminds the interviewer of the subject areas. In this way the interviewer can establish a conversational style (Patton, 2002). The interview questions mainly concerned:

- The tasks they carry out
- The decisions, judgements, and trade-offs they make
- The problems and difficulties that face them

We also tried to expand the focus area by asking more about aspects that did not just verify what was already known. This means we paid attention to unexpected aspects and thereby tried to avoid getting caught in false or too narrow presumptions which we could have.

Each interview lasted between 1.5 and 4 hours. Two of the interviewees were interviewed together, because they wanted it that way. All interviews except one were recorded and transcribed. The excepted interview was somewhat different from the others, in the sense that it also included many more background questions, for example, concerning the company. An interview guide was used in all interviews, but the four-hour interview had a more informal and conversational nature than the others. Furthermore, during the interview notes were taken.

The data collection techniques worked well and the samples were information rich. Thanks to the recordings of the interviews, the reflection on our own behaviour was facilitated. We have not detected any leading questions, and we tried to expand the focus area as prescribed by Holtzblatt and Jones (1993). However, we did not always manage to do that, which became clear during the data analysis. As a consequence, it has not been possible to provide elaborate descriptions of the decision outcomes.

Data analysis I

After the interviews with the requirements engineers, the recordings were transcribed into written documents and a content analysis was conducted. Content analysis is the sense-making effort in which patterns or themes are identified (Patton, 2002). Some of the analytical tools and coding procedure provided by grounded theory (Strauss & Corbin, 1998) were used, particularly during the analysis of factors that affect the RE decision-makers. The decision situation framework provided high-level categories, which offered a structure to the data analysis process. Within the framework, the lower-level categories emerged from the empirical data, and these categories formed the basis for the description of the decision situation of RE decision-makers. The analysis was performed in four steps.

1. The categories from the decision situation framework were used while reading the transcripts thoroughly. Sentences and sections were marked as
belonging to a certain category in the framework. The purpose was to identify all pieces of relevant information concerning the decision situation in the documents.

2. Short sentences were made for each mark, and each short sentence was listed in its framework category.

3. The short sentences were grouped and conceptualised. This means that categories emerged from the data, which constituted the foundations of the resulting findings. These were documented in a table that was organised according to the categories.

4. The combined decision process model (see Figure 25 on page 110) was used to reconstruct the decision processes. Furthermore, the decision-making activities, identified in the empirical data, were connected to the routines in the decision processes. In addition, we used the supporting communication routine in Mintzberg et al. (1976). Finally, the processes were documented in tables.

Subsequently, the result from this analysis was reported and discussed in a focus group session.

**Focus group**

According to Yin (1994), triangulation can be used to strengthen a study. One way of triangulating is to use methodological triangulation, which means using multiple methods to study a single problem. There are two forms of methodological triangulation; within-method triangulation and between-method triangulation. Within-method triangulation concerns the use of multiple strategies within one method, e.g., different measuring scales in a survey questionnaire. Between-method triangulation means combining different methods (Denzin, 1978). Denzin (1978) advocates the latter, since it balances the weaknesses of a method with the strengths of another. Therefore, a focus group was used as a data collection method to complement the interviews.

The focus group session was held with nine participants at the company. Two of the interviewed requirements engineers were also present. The other participants included systems engineers, a product manager, a systems manager, requirements engineers, and subsystem test engineers. The result from the analysis of the interview data was presented in the first part of the session, and, in the second part the participants discussed the result and commented on it. During the session, which lasted for two hours, the discussion was recorded and subsequently transcribed.

**Interviews II**

In the second round of interviews, five persons associated with the requirements engineers were interviewed. In this way it was possible, to some extent, to achieve data triangulation. According to Denzin (1978), data triangulation is when a variety of data sources is used, such as dissimilar groups or settings. In this research project
the former way is used. The variety of data sources lies in the different roles of the interviewees between the first and the second round of interviews.

We used the snowball sampling technique to locate information-rich informants. Snowball sampling means the researcher asks people he or she meets about relevant and knowledgeable individuals, who in turn are asked about other informants and so on (Patton, 2002). The interviewees comprised a system manager, a subsystem test manager, a subsystem manager (who also worked as a software engineer), and two system engineers, one of whom also worked as a software engineer. Two of them had participated in the focus group, and had between three and fifteen years of relevant work experience.

As in the first interview round, open-ended interviews, inspired by contextual inquiry, were used. The questions were not standardised. The interviews had a conversational style and we used an interview guide. The interview questions mainly concerned:

- The tasks of the interviewees
- The tasks of requirements engineers
- The decisions, judgements, and trade-offs made by requirements engineers
- The problems and difficulties related to the requirements engineers

These interviews were also conducted at the workplace of the interviewees and each interview lasted between 1 and 1.5 hours. Two of the participants were interviewed together, because they wanted it that way. All the interviewees were first asked to describe their own work and then to discuss the work carried out by the RE decision-makers, as well as problems and difficulties related to that. The interviews were recorded and transcribed.

**Data analysis II**

In the second data analysis the results from the first data analysis were compared to the new data, i.e., from the focus group as well as the second interview round. During the comparison the preliminary results were verified and nothing was removed. The second data analysis provided more nuance and deepened the understanding of the context in which the RE decision-makers act as well as the decision situations. One factor and one information source were added at this stage, and more relationships between factors were identified. In this data analysis round, we also explicitly used external theories as prescribed by multi-grounded theory (Goldkuhl & Cronholm, 2003). The empirical results were thus related to relevant decision-making theories.

The empirical findings showed that there are two main RE decision processes; a) establishment of requirements and b) management of requirements changes. We had rich information of the first decision process, i.e. the establishment of requirements. However, we did not have equally rich information about the second decision process, i.e. managing requirements changes. We also needed more information about the experiences of using RE tools, since our research aims towards improving
the decision-supporting capabilities of RE tools. Thus, we needed to do complementary interviews.

**Interviews III**

In the third interview round, five participants were interviewed. Just as in the previous interview rounds purposeful sampling was used. Some individuals were found via our contact person at the company and others were found using snowball sampling. The interviewees comprised two requirements engineers, a section manager, a system manager, and a system engineer. The requirements engineers had been interviewed in the first interview round, and had between five and thirty years of relevant work experience.

As in the previous interviews, we used an open-ended interview technique and the questions were not standardised. Instead, an interview guide was used together with a conversational interview style. The interviews consisted of questions concerning:

- The requirements change management process
- The work in relation to requirements change proposals
- Requirements management tools

These interviews were also conducted at the workplace of the interviewees and each interview lasted between 0.5 and 1.5 hours. All the interviews but one were recorded and transcribed. There was a technical recording problem with one interview, which fortunately was discovered early in the interview. During this interview, notes were taken, which together with the transcripts were analysed with previous data.

**Data analysis III**

In the third data analysis, we compared the new data with the previous data focusing on requirements change management. Furthermore, the previous results were verified and nothing was removed. The third data analysis provided a more nuanced and deepened understanding of the requirements change management aspects of the RE decision process. We also gained more insight into the use of RE tools which we kept in mind when developing the REDSS characteristics and guiding principles.

This data analysis showed that we had reached enough theoretical saturation for the characterisation of the decision situation of RE decision-makers. Consequently, we decided to discontinue the data collection at the case study company. Instead, we wanted to obtain some indication of the transferability of the case study results, and, for that reason, conducted transferability interviews at another company.

**Transferability interviews**

As we previously indicated, it is not possible to generalise from the observations of one case study. However, the potential transferability of our results to other organisations can be investigated. Therefore, we conducted interviews with experienced consultants working at a Swedish IT consultancy company. Our contact person at this company suggested four people who fulfilled the profile we wanted,
i.e., IT consultants who have experience of and/or have a good insight into the work of requirements engineers from several different organisations. They had between 10 and 25 years of relevant work experience.

We conducted two open-ended group interviews in which we used non-standardised questions. We chose group interviews, since we assumed that discussions between two persons would be more information-rich than one on one interviews. We applied a conversational interview style, and the interviews lasted for two hours and were held at the workplace of the interviewees. In the interviews, the interviewees were provided with paper-based PowerPoint slides of our characterisation of the decision situation of requirements engineers. Due to time limitation, we only included the two RE decision processes, their decision-making activities and decision matters, and the factors that affect RE decision-makers. We presented one or two slides at a time and then discussed the content with the IT consultants. The discussions were driven by questions such as:

- What agrees with your experience?
- What disagrees with your experience?
- To what extent does this agree with your experience?
- What is missing?
- What agrees sometimes and what does not agree sometimes?
- What can cause the differences?

The interviews were recorded and transcribed. By comparing the data with the case study results we obtained some indication of the transferability. The indication is that the results largely agree with the experiences that the IT consultants have of other types of organisations, although there are also aspects that differ between companies. Thus, we concluded that the results of the case study are partially transferable to other organisations.

The results from the case study were then used in the synthesis, where desirable characteristics and guiding principles of a visionary REDSS were developed.

### 5.2.4 Synthesis – development of desirable characteristics of an REDSS

In the synthesis, the empirical results were merged with relevant theories and techniques, e.g., related to decision-making and techniques from decision support systems, in order to develop desirable characteristics and guiding principles, as well as providing examples of available techniques. The purpose of the guiding principles and available techniques is to direct further efforts concerning how to find solutions that fulfil the desirable characteristics. Together, the desirable characteristics and their guiding principles constitute a visionary REDSS, which is tied to practical applicability via the available techniques. While the desirable characteristics are only empirically grounded, the guiding principles are empirically and theoretically grounded and the available techniques are only theoretically grounded.
The desirable characteristics are based on the findings in the case study in terms of the needs of the RE decision-makers and on the nature of the tasks in the RE decision processes. The characteristics are not based on specific RE activities and tasks. This means we do not directly address RE decision matters such as choosing the most appropriate elicitation technique or prioritising requirements. Instead, we focus on the generic human activities that take place in the RE decision processes, for example, problem solving, communication, and idea generation. These generic human activities are found in the specific RE decision matters. The characteristics represent how an RE decision-maker will experience the REDSS.

The characteristics are categorised at different levels of support: 1) the RE decision-maker as a user, 2) the nature of RE decision-making tasks, and 3) the RE decision-maker in the social context. These levels coincide with the conceptual model for human-computer interaction (HCI) by Eason (1991). A support system can only be useful in relation to the characteristics of the target users, the tasks to be carried out, and the context in which the system will be used (e.g. Maguire, 2001). It is very important to adopt a user-centred design approach to obtain a successful decision support (Parker & Sinclair, 2001). Thus, these levels suit the purpose of proposing characteristics of REDSS and make clear that all levels are, at least to some extent, covered.

Like the desirable characteristics, the guiding principles are empirically grounded. However, contrary to the characteristics, the principles are also theoretically grounded, which means we base our suggestions on the case study findings as well as theoretical knowledge. The purpose of the guiding principles is to direct further efforts concerning how to find a solution. For extensive information of the derivations of the characteristics and guiding principles, see section 9.1.

The available techniques represent a way of elaborating the guiding principles and to show that it should be possible, in some sense, to realise the visionary view of REDSS. We searched for existing solutions and principles in the fields of human-computer interaction, information visualisation, and decision support systems. These fields are appropriate since we have a human-centred, a decision-maker-centred perspective. We have not searched for optimal solutions, for two reasons. A) Ngo-The and Ruhe (2005) argue that RE decision support should not strive for optimality, since many decision situations in RE are not simple enough to enable an absolute optimal solution. B) We did not have instruments and time to determine if a solution is optimal or not. Instead, we explored the fields until we found a technique that seemed suitable for the guiding principle at hand. The techniques that are presented together with the guiding principles should be viewed as a range of potential means.

Hence, the characteristics of REDSS and the characteristics are based on the empirical findings of the case study. For that reason, the empirical part of the research process is critical.
5.3 Reflections on the empirical part of the research process

From a quantitative point of view, the qualitative research approach has limitations. Shaughnessy and Zechmeister (1997) stress that scientific control is the essential ingredient and what needs to be controlled are independent variables. In that regard, qualitative research is unscientific. Obviously, we do not agree with Shaughnessy and Zechmeister (1997). Their characterisation of scientific and non-scientific approaches applies to, what Starrin et al. (1991) call, the context of justification, which includes hypothesis, operational definitions and hypothesis testing. However, the context of justification is not the only way. Another way is the context of discovery that results in the formulation of theories or hypotheses. In the context of discovery; creativity, intuition, and sensitivity are important (Starrin et al., 1991). This research project follows the way of discovery. It has, for instance, resulted in descriptions of the decision situation of RE decision-makers and desirable characteristics of an REDSS. These have been generated but not proved. The context of discovery was necessary in this project since there does not yet exist a cohesive body of knowledge on RE decision-making and RE decision support. When such a body of knowledge has evolved, the context of justification will be the most appropriate way.

The quality criteria for quantitative research are not appropriate for qualitative research. “The quality criteria for statistical surveys, such as reliability and representativeness, cannot be applied to case study research, nor to any other of the approaches described below [e.g. grounded theory, ethnography, action research]. They are not scientifically general” (Gummesson, 2001). Lincoln and Guba (1985) argue that the appropriate criteria for trustworthiness in qualitative research are credibility, transferability, dependability, and conformability.

Credibility

Credibility is the substitute for internal validity and denotes that the reconstructions, i.e., the outcome of the research process, are “credible to the constructors of the original multiple realities” and that the inquiry is carried out in a trustworthy way (Lincoln & Guba, 1985, p 296). This means that the persons in the case should recognise the reconstruction. For example, the interviewees should recognise our description of the RE decision processes and the factors that affect the RE decision-makers, although they have not explicitly discussed them in these terms during the interviews. They should feel comfortable with our interpretations and conceptualisations. In the case study, we verified the results twice in the company. Firstly, in the focus group session that was also used for data collection purposes, and secondly, in a presentation at the company, where, e.g., requirements engineers and higher-level managers participated. The purpose of the second presentation was primarily to inform those who work with process improvement at the company. The case study results were found credible by the persons who participated on those two occasions.

Lincoln and Guba (1985) claim that credibility is strengthened through techniques such as prolonged engagement, persistent observation, triangulation, peer
debriefing, negative case analyses, referential adequacy, and member checking. Efforts to triangulate were made, for example, methodological triangulation in the form of different data collection techniques and data triangulation in the form of different roles of the interviewees. While it is preferable to triangulate, it would perhaps have been even more preferable if more different data collection techniques and data sources had been used. It would probably be informative to carry out a longitudinal observation of RE decision-makers’ daily work or ask them to write diaries, but for practical reasons this was not possible in our case.

Lincoln and Guba (1985) note that there can be distortions stemming from the interviewees that affect the data quality and thereby the credibility. During the data collection of this research process two specific possible distortions were identified. The first was the group interviews in which two persons were interviewed at the same time at their own discretion. There may be a risk that they influenced each other, e.g., things they wanted to say but for some reason chose not to. However, they did not appear to avoid discussing somewhat delicate aspects in the group interviews, and they did not always agree. It was rather the opposite, they triggered each other and the discussion was rich. Furthermore, neither of the two interviewees in the group interviews was particularly dominant. The second potential distortion was that two of the interviewees in the second interview round of the research process had participated in the focus group. While this may have influenced their answers, we did not explicitly discuss the preliminary results that had been presented in the focus group session. However, the interviewees did refer to the results a couple of times.

In order to reach credibility, the researcher has to identify the important aspects and sort out the irrelevant aspects (Lincoln & Guba, 1985). During the data analysis, the decision situation framework provided support in several ways compared to using no framework at all. In a broad sense, it enabled a more complete view of the decision situation. The framework provided predefined high-level categories, where each category is represented by a “box” in the framework. These predefined categories made it possible to, in a structured way, walk through the data category by category, and thus the risk of missing important aspects of the decision situation was reduced. When we found that there was not enough information concerning a certain aspect, e.g., decision outcome, it was possible to collect complementary data pertaining to this aspect. The framework enabled the gathering of extensive information within each aspect of the decision situation. The framework directed our attention to aspects found in the data that we believe could have been more easily overlooked without the framework. It forced us to identify actual decision activities as well as find the real problems and difficulties facing the decision-makers, instead of identifying, e.g., organisational procedures and information flows, which are – of course – important, but not enough. The decision-maker perspective of the framework made us focus on the needs and requirements of the RE decision-makers.

Although, as argued above, the framework was supportive. There is a risk that it and the preceding literature analysis made us less open-minded to the empirical data, which is one of the underpinnings of grounded theory. To put it in Strauss’ and
Corbin’s (1998, p 49) words, “the researcher does not want to be so steeped in the literature that he or she is constrained or even stifled by it”. Conversely, this position is criticised by Goldkuhl and Cronholm (2003). They argue that if the researcher is un-prejudiced in data collection and data analysis, then there is “a risk of being naïve and even ignorant when entering the empirical field (Goldkuhl & Cronholm, 2003, p 3).

In summary, we consider the results of the case study to be credible. Efforts to strengthen the credibility have been made during the whole inquiry and the persons at the case company feel comfortable with our interpretations and conceptualisations. Although, given more time there are more techniques that could have been used in order to further strengthen the credibility.

**Transferability**

Transferability is related to the concept of external validity, but Lincoln and Guba (1985) argue that in a strict sense it is impossible to reach external validity. Instead, the researcher has to provide a detailed description of the context in which a working hypothesis is found to hold. The transferability is then connected to the degree of correspondence between the sending and receiving contexts. While the sending context is the case from which the conclusions are made, the receiving context is the case that may use the results. Moreover, the judgement of transferability lies with the person who wishes to use the results (Lincoln & Guba, 1985). This means that we should provide information that facilitates an assessment of the transferability. In order to facilitate the assessment, we described the research context of the case study and conducted interviews to get indications of the transferability.

In our case, the transferability interviews show that the decision situations of RE decision-makers in the case study are not unique and has transferability potential.

**Dependability**

There can be no credibility without dependability. In qualitative research, dependability replaces the notion of reliability in quantitative research. In research designs such as ours, it is not possible to replicate the inquiry, since it is not possible to “cross the same stream twice” (Lincoln & Guba, 1985, p 299). However, it is important to give a careful account of the research process and the methodological considerations, which we in the previous section have tried to provide.

Lincoln and Guba (1985) propose using techniques with regard to dependability such as overlap methods and inquiry audit. The overlap methods technique is a type of triangulation and has been used in this research project as described and discussed above. The technique inquiry audit is concerned with examining the research process and the research product. This has been done through exposing the process and product by submitting our research to and attending peer reviewed conferences and workshops (Alenljung & Persson, 2004; 2005a; 2005b; 2005c).
We consider the result from the case study to be dependable, since we have been paying attention to aspects that increase the dependability of the results throughout the research process.

**Conformability**

The last of Lincoln and Guba’s (1985) quality criteria is conformability, which is the qualitative term for objectivity. Conformability is not concerned with the objectivity of the researcher; instead the focus is on the quality of the report. It should be possible to audit the process and trace between the raw data and the final outcome, so that it is possible to see the correspondence between the results and the empirical data.

It has been our intention to be transparent and achieve traceability when reporting the process and the findings. We tried to elaborate the details of our research process in order to facilitate an audit of the process. We also frequently provided quotations from the empirical data, so that it to some extent should be possible to assess our interpretations of the data. Thus, we consider the results to be conformable with their empirical origin and that it is possible to judge the conformability from our report.

**5.4 Chapter summary and reflections**

The nature of the research problem made it suitable to use a qualitative research approach as well as a design science approach. The research process was conducted in three steps. First, a literature analysis was made, which resulted in a generic decision situation framework. Second, a case study was carried out at a systems engineering company. In the case study, requirements engineers and actors associated with them in the systems engineering process were interviewed. A focus group session was also carried out. The case study resulted in a description of the decision situation of RE decision-makers. Third, a synthesis between our empirical findings and theoretical aspects was made in order to develop desirable characteristics of a visionary REDSS, guiding principles and to put forth some available techniques.

We consider the results to be trustworthy, since credibility, transferability, dependability, and conformability have been taken into account during the research process as well as in the report of process and findings. The main positive aspects of the research process of the case study are that:

- Method and data triangulation were used
- The high level framework provided support to the data analysis
- Transferability interviews were carried out
- The process and results were exposed to peers
- There is transparency of the research process
- There is frequent integration of citations in the presentation of the results
The main negative aspects are that the methods and data sources may not have been different enough, and that the framework and its theoretical foundations may have been too much of an influence during the data analysis.

The research methods were appropriate, since they made it possible to adequately address the research problems. The research process has made the results empirically as well as theoretically grounded. There are not any specific parts of the process that we can perceive could have been carried out differently considering the time constraints. Nevertheless, time limitations forced us to focus on parts of the research cycle. Hence, we have not been able to validate the results.

The work in this research project has resulted in several contributions, the first of which is the decision situation framework.
6 Decision situation framework

In this chapter, the generic decision situation framework is described. Our working definition of a decision situation is that it is a contextual whole of related aspects that concerns a decision-maker. The proposed framework provides a holistic view of the decision situation and it can, for example, be used as a guide when analysing a decision situation at hand during early phases of decision support development. In the following, the framework is presented in its entirety.

6.1 Description of the decision situation framework

The framework in Figure 24 is the result of a thorough analysis of decision-making literature. It describes important aspects related to a decision-maker and demonstrates the complexity of decision-making.

Decisions can be made by different classes of decision-makers: a) an individual decision-maker, b) a team or c) a group. Even if the decision-maker is an individual, he or she is also often a part of a group or an organisation. This in turn, is part of the context in which the decision-maker acts and the decisions of interest are often related to an organisation.

The decision-maker deals with a decision matter. A decision matter can, for example, be a reorganisation of a company, a new marketing campaign or a new investment.
Such decision matters are not just a choice between alternatives, but several more steps within a decision process are needed.

The decision process can be divided into two parts, a “pre-choice” part and a “post-choice” part. The pre-choice phase includes, for example, understanding the problem, generating alternatives, evaluating alternatives, and this process ends when the decision-maker makes a choice, i.e., results in a decision. This decision is viewed as the outcome of the decision process. The decision is then implemented and perhaps later on also followed up. As a complement to the decision situation framework, we suggest the decision process model illustrated in Figure 25. For descriptions and motivation see section 3.3.3.

![Figure 25, A decision process model, combined from the models of Mintzberg et al. (1976) and Power (2002)](image)

A decision process consists of several decision-making activities which can be described by decision theories. There are two types of descriptive theories of individual decision-making, the “traditional” theories that are based on laboratory studies, i.e., Judgement and Decision-Making (JDM) and theories based on studies made in natural environments, i.e., Naturalistic Decision-Making (NDM). JDM theories focus on how people make choices from a set of alternatives, for example, how decision-makers use heuristics in complex situations and which biases, i.e., patterns of errors, they can lead to (see e.g. Tversky & Kahneman, 1974). Studies within NDM, which are made in a natural context, aims to describe how decisions are made in the natural environment (see e.g. Orasanu & Connolly, 1993). The discipline of group decision-making (GDM) addresses questions concerning what characterises decision-making with multiple participants, what kind of problems they face and how group decision-making can be improved (Miner, 1992). Organisational decision-making (ODM) is studied from different approaches, e.g., a process-oriented view, an organisational
procedures view, and a political view (Keen & Scott Morton, 1978). The process-oriented perspective of ODM describes how decision-making in organisations is actually carried out and what goals are used. The organisational procedures view concerns the formal and informal structures of the organisation, organisational roles, procedures, and communication channels (Keen & Scott Morton, 1978). From the political approach the bargaining process is important, and it also focuses on power and its influence on the decision (Keen & Scott Morton, 1978).

There are three aspects that affect the decision-maker’s behaviour when carrying out decision-making activities. These comprise: a) the characteristics of the decision-maker, b) factors, and c) information. The decision-maker has characteristics, such as knowledge, experience, personality, and cognitive abilities, which affect how the decision-maker carries out the decision-making activities. While some characteristics, such as psychological types and decision styles, are individual, other characteristics, such as degree of expertise, can be identified for a certain group of decision-makers.

The decision-maker’s behaviour in decision-making activities is also affected by factors that originate from the context. There is ambiguity which may be caused by ill-structured problems; uncertain, dynamic environments; shifting, ill-defined, or competing goals or values; ambiguity of information; and interpretation of the history of decisions. In addition, there are multiple players who can be involved in conflicts. Repeated decisions and a longitudinal context that call for action and feedback loops also exist. And finally, there can be organisational goals and norms, time stress, and high stakes.

Another aspect that can affect a decision-maker’s behaviour and the choice that is made is the information that comes from different sources. The decision-maker receives input, e.g., information from a database or ideas from a colleague that he or she can use. This input has characteristics, e.g., data quality.

By including all these aspects, we can describe a contextual whole of related aspects that concerns a decision-maker, i.e., we can give a picture of the decision situations of decision-makers.

6.2 Chapter summary and reflections

In this chapter, we describe the generic decision situation framework. The decision situation framework is a theoretically based framework that takes a holistic approach of the decision situation from the viewpoint of a decision-maker.

The framework facilitates qualitative analysis of empirical data when studying decision-making in a particular context. It may also be supportive in a quantitative research setting, for example, structuring the research design. However, in this research project, it has only been used in a qualitative research setting. Thus, we can only claim usefulness for qualitative analysis. The framework supports the qualitative analysis in three main ways compared to using no framework at all: a) it enables a more complete description of the decision situation; b) it enables the
gathering of extensive information within each part of the decision situation; and c) it forces us to identify actual decision activities as well as finding the real problems and difficulties facing the decision-makers.

In a broad sense, the framework enables a more complete description of the decision situation, and provides predefined categories, where each category is represented by a “box” in the framework. These predefined categories make it possible, in a structured way, to walk through the data, category by category, which reduces the risk of missing important aspects of the decision situation. If the analyst finds that there is not enough information concerning a certain part, e.g., decision outcome, it is possible to collect complementary data on this part. This was exemplified in the case study when missing information concerning the decision outcome was located during the data analysis. Following the case study analysis, it became obvious there was not enough information concerning decision outcome. While ten decision matters and two different decision processes were found in the case study, it was not possible to state in what way each of these was expressed. For example, the decision matter ‘How should the particular requirements change be managed?’ had no related decision outcome in the data. Furthermore, questions were raised such as: are these decisions documented and has information concerning the decision been distributed to others? The interviewees had described in what way the main decisions were expressed. By main decision we mean the last decision made in each decision process. The main decisions can also be seen as defining what the decision-makers’ work “is all about”. However, the outcomes of all sub-decisions, which preceded the main decision, were not in the data. We had to make complementary interviews to “fill in the gaps”. If we had not analysed our data in terms of both decision matter and decision outcome, which is prescribed by the framework, we doubt that we had observed the discrepancy. Thus, a more complete picture was gained.

The framework enables the gathering of extensive information within each part of the decision situation, by directing our attention to aspects found in the data that we believe are more easily overlooked without the framework. This was exemplified when decision matters that were not explicitly stated as decisions by the interviewees were identified. During the analysis we found that there were more actual decision matters than explicitly articulated by the interviewees. The decisions that they claimed they made were of a ‘final’ nature. However, in our case (and probably in most cases) there were decisions preceding the main decision, which we call sub-decisions. Some of these sub-decisions, for example, ‘Which system requirements should belong to which subsystem?’, were “hidden” in the data as activities. The interviewee spoke about them as “things they do”. The given example was, in fact, an important tactical decision negotiated between several decision-makers. Thus, this information was not found directly because of the questions asked during the interview session, but because the data was analysed in terms of decision matters, as prescribed by the framework. Consequently, more thorough information concerning the category was gained.

The framework forces us to identify actual decision activities as well as to find the real problems and difficulties facing the decision-makers, instead of identifying, e.g.,
organisational procedures and information flows. Organisational procedures and information flows are, of course, important, but they are not enough. The decision-maker perspective of the framework makes us focus on the needs and requirements of people who are going to use the decision support system (DSS).

Using the decision situation framework we can portray a particular decision situation of a certain decision-maker in a human-centred and holistic manner. In this thesis the decision situation of RE decision-makers is in focus.
7 Decision situation of RE decision-makers

In this chapter, we present a) decision matters, decision-making activities, and decision processes in RE, b) characteristics of decision matters, as well as c) information sources used by RE decision-makers. The focus area is RE decision-makers on a subsystem level in a contract development context, where both projects and systems to be developed are large and complex.

The case study shows there are two separate RE decision processes. The first decision process concerns the establishment of requirements, and the second concerns management of requirement changes. These two processes have similarities, but also important differences from a decision-making perspective. Furthermore, both processes include a number of decision-making activities in different decision phases. Both are highly iterative and embrace several sub-decisions, which are work-related and system-related, and affect the efficiency of the systems engineering process as well as the quality of the product. The RE decision-makers use different information sources as input, for instance, requirements, formal as well as informal reports, and written as well as verbal sources.

Direct quotations are included in the text for illustration, since “description and quotation provide the foundation of qualitative reporting” and the report should provide “enough detail and evidence to illuminate and make it that case” (Patton, 2002, p 503). A quotation is, by definition, in the exact words of the person who said it. However, in this thesis all quotations are translated from Swedish. Nevertheless, in order to be true to and represent the interviewees own experiences and their own terms, we tried to translate as close as possible to the original quotation as possible. Consequently, the English translations can appear slightly “weird”. The original citations are listed in Appendix, and the quotations are highlighted with the help of indentation and font size. In addition, the interviewee profile is given in parenthesis after each citation.

7.1 Decision matters, decision-making activities, and decision processes in RE

The decision-making activities in the two identified decision processes in RE have been structured using the phases and routines in the decision process model of Mintzberg et al. (1976), complemented with the last two phases of the decision process model of Power (2002) (see Figure 25 on page 110). With this addition both the pre-choice and post-choice activities are represented in the decision process. Mintzberg’s model also provides three sets of supporting routines: a) decision control routines, e.g., allocating organisational resources, b) decision communication routines, and c) political routines, e.g., clarifying power relationships in the organisation. In the empirical findings there are no activities that can be categorised as decision control and political, at least not in the way Mintzberg et al. (1976)
describe them. However, activities that can be categorised as decision communication frequently occur in the empirical findings, and are therefore included in the presentation below. There are three decision communication routines: a) the exploration routine, i.e., “general scanning for information and passive review of what comes unsolicited”, b) the investigation routine, i.e., “focused search and research for special-purpose information”, and c) the dissemination routine, i.e., disseminating information of the progress of the decision process and the decision outcome (Mintzberg et al., 1976, p 261).

7.1.1 Establishment of requirements
This decision process concerns the establishment of requirements. Hence, it occurs early in the project. Although, the description of the decision process gives it a sequential appearance, this is incorrect. Since the process is highly iterative, each routine can be activated several times and in different orders. Hence, each instance of the decision process can take any path through the routines.

The tasks conducted in this process are summarised by an interviewee as can be seen in Quotation 1 and Quotation 2:

Quotation 1: “… it is actually about getting some form of input that is to be structured in some way and written down on this little “box” [i.e. a computer], that we are to make.” (Requirements engineer)

Quotation 2: “What you do as a requirements engineer is receive requirements, write them, analyse what matters for us, then discuss with the user [the interviewee uses the term ‘user’ for several types of stakeholders and not just for end user] and check […] that you have understood it correct…” (Requirements engineer)

The identification phase
The decision process begins with the decision recognition routine. The decision process termed ‘establishment of requirements’ is activated when the subsystem RE decision-maker receives new customer and system requirements from the “system level”.

In the diagnosis routine, “the decision maker is faced with an array of partially ordered data and a novel situation” (Mintzberg et al., 1976, p 254). In this routine, the RE decision-maker conducts several decision-making activities: find out what the customer requirements and system requirements mean (see Quotation 3), investigate ambiguities in system requirements (see Quotation 4), initiate themselves into interfaces provided by the customer (see Quotation 5), analyse what matters for the subsystem (see Quotation 6) and perform a basic analysis of the desired functionality (see Quotation 7). While all of these activities are system-related, there is also a process-related activity, in which the RE decision-maker creates a general view of the needs and problems in the development process to come (see Quotation 8).

Quotation 3: “… a requirements engineer should of course be good at his own subsystem, but he must also have an understanding of our whole system to
know what subsystem is surrounding it. [...] And if you know what our system looks like and what it is going to be used for then you will make faster and more correct decisions. [...] besides making your own requirements specification and having control of it, you have to understand the system’s requirements specification because it is from there you pull out the requirements for the subsystem together with the requirements allocation that we also do then.” (System engineer manager)

Quotation 4: “Well, yes, in principle there are a lot of investigations. If we find something ambiguous in the system requirements, then we have to talk to the customer to find out what they really mean. I think that the major part of my work is about making investigations.” (Requirements engineer)

Quotation 5: “We have a lot of communication protocols to [another company’s units]. Other old [X] systems and a communication computer that a consultancy company has made available that belongs to the customer. All such interfaces are made available by [another organisation]. Then we have to initiate ourselves into these.” (Requirements engineer)

Quotation 6: “You specify what requirements that belong to which subsystem, so that all subsystem managers and requirements engineers for the subsystems know which requirements they should fulfil. [...] You had to read it [i.e. the system requirement specification] through a number of times and try to figure it out. This sounds like our responsibility. What you do next is to ponder. What is implemented so far? What does the structure look like? How can we include this functionality in the existing set of requirements and construction? Will some discrepancy appear? Can there be problems further on? You simply have to sit down and think about how it is intended to work.” (Requirements engineer)

Quotation 7: “… investigate functions. What are going to be there? Often, it is not the case, unfortunately, that it is only ‘what’, but it is also a lot of ‘how’. It is unavoidable to end up in this how discussion at the same time.” (Requirements engineer)

Quotation 8: “But there it is also really important that you… that you have a common picture of what is the difficulties in the project. So that you really find out what is really important with this.” (Focus group participant)

Two decision communication activities are identified in the diagnosis routine. The RE decision-makers conduct investigations in which they obtain an understanding of the problem by searching documents (see Quotation 9) and talking to relevant stakeholders (see Quotation 10), e.g., the customer. The dissemination that takes place is to notify those who are responsible for the entire system when there are problems in the customer requirements (see Quotation 11).

Quotation 9: “Some solve problems by discussing with others and finding themselves a contact net and solve it that way. Others maybe sit in their room and go through... maybe read through this requirements specification on the level above from the first page to the last page...” (Requirements engineer)
Quotation 10: “Sometimes the whole investigation is a discussion between him and me for example.” (Subsystem manager and software engineer, i.e. one person with two roles)

Quotation 11: “Then I had to go through the protocols and try to figure out what they really mean. And I found no way to transfer [X to Y]. Then I had to notify the system level about it and they contacted the customer.” (Requirements engineer)

The development phase

In the search routine, the decision-maker seeks ready-made solutions and in the closely related screen routine these alternatives are reduced to a few feasible ones (Mintzberg et al., 1976). In these routines the RE decision-makers compare the new requirements with existing components and find out if something can be reused (see Quotation 12).

Quotation 12: “Instead, all the requirements engineers receive, so to speak, is this system specification and then it is up to us to discuss it. So actually what this is about, based on the requirements specification that comes from above, is to extract which are the requirements for our “box”. You make this agree with the history we have in the form of other constructions with similarities to ours and try to reuse them and a bit like that.” (Requirements engineer)

In the design routine, custom-made solutions are developed and ready-made solutions are modified (Mintzberg et al., 1976). The RE decision-makers create use cases and write requirements (see Quotation 13), such as internal requirements and requirements that specify the interface between the subsystems (see Quotation 14). Dependencies between use cases are also drawn in this routine (see Quotation 15).

Quotation 13: “You write quit a lot about what the product is going to look like. […] so you have to by yourself construct your system at a high level of abstraction to a large extent, as I see it. […] Of course, we get help from others and do a lot of investigating, but basically it is the one who writes the requirements that puts it all together to a system.” (Requirements engineer)

Quotation 14: “This is now divided so that we should write a whole bunch of requirements that specify the interfaces between the subprojects. […] It becomes lot of requirements that have to be agreed upon among the subsystems.” (Requirements engineer)

Quotation 15: “So, unfortunately you cannot just write requirements directly from your mind. […] Well, you can do that in the beginning of a project. You chat about it with the subsystem manager. Finally, you have some idea of what it is going to look like. Then you sit down and… You have probably already drawn some use cases and what dependencies there will be between them. So, you create them, write requirements atoms, do a check.” (Requirements engineer)

Three decision communication activities are identified. All three are categorised as belonging to the investigation routine. The RE decision-makers discuss ideas and solutions with those who are responsible for the entire system (see Quotation 16).
They also sometimes discuss with other people who are responsible for subsystem requirements (see Quotation 17 and Quotation 18). Each person documents the result of these discussions in their “own” use cases. They also have to stay alert to the customer requirements and system requirements, so that these are covered in the subsystem requirements (see Quotation 19).

Quotation 16: “But I think that it is important to have communication all the time with the different concerned instances then. What is important and what is less important? Try to make sure that you get what the customer wants and concentrate on that, instead of making up requirements cases on your own.” (Subsystem test manager)

Quotation 17: “I don’t have so much contact with the other subsystems. The only contact I have is with the subsystem manager in the project, I mean the subproject, with whom I discuss the ideas. Maybe I suggest a solution and then he says ‘oh no that is not possible because then, you know, we have to rewrite this component’.” (Requirements engineer)

Quotation 18: “Which information is needed here? Then perhaps you realise that you have to cooperate with the other subsystems. It may be required that a position is sent. Then you cannot just write a requirement. Then we have to inform [subsystem X] that they have to provide this position.” (Requirements engineer)

Quotation 19: “But as a requirements engineer you have an informal responsibility upwards to bring all requirements downwards, but you have a responsibility downwards to make sure that all requirements are brought into the requirements specifications that the software engineers or he who designs the subsystem then. You have a formal responsibility downwards, but an informal responsibility upwards.” (Requirements engineer)

The selection phase

The evaluation-choice routine consists of three different modes: analysis, bargaining, and judgement. In the analysis mode, the alternatives are evaluated (Mintzberg et al., 1976), and the RE decision-makers trace the requirements to higher level requirements (see Quotation 20). They check the requirements together with other stakeholders (see Quotation 21), and also analyse risks together with others (see Quotation 22). The decision communication carried out is to call together everybody concerned to a requirements check (see Quotation 23).

Quotation 20: “Then we also try to spend time on some requirements tracing for as long as it is possible, so to speak. To really connect them, and that we also do in the database then, the requirement on the system level to the requirements we have on the subsystem level. That you can follow and generate a print-out of it also, see that this system requirement has become this and that. This is something that is very easy to say that you should do, trace requirements. However, you soon land into a grey zone, so to speak.” (Requirements engineer)

Quotation 21: “We have checks between the subsystems. When I have written down my requirements, I call a check meeting. […] Then we walk through the
In the **bargaining mode**, there are several decision-makers with different goals that make the choice (Mintzberg et al., 1976). Two system-related decision matters are dealt with by the RE decision-makers (see Quotation 24):

- Which system requirements belong to which subsystem?
- Which actors are there (see Quotation 25)?

In the **judgement mode**, an individual makes the choice (Mintzberg et al., 1976). Three system-related decision matters are handled by the RE decision-makers:

- Can the requirements become baseline?
- How is the subsystem going to behave and what is it going to look like?
- Which use cases are needed (see Quotation 26)?

There are also work-related decisions that the RE decision-maker makes. These are:

- Which level of detail should the requirements have (see Quotation 27)?
- What type of information should the requirements contain?
- In which order shall the requirements be implemented, i.e., what is the priority of the requirement (see Quotation 28)?
- Which level of effort should an investigation have, e.g., of functions in order to write requirements (see Quotation 29 and Quotation 30)?

Quotation 24: “I make a lot of decisions. But at the same time, since we check everything and like that, so are all things such as what you write and how you write, and things like that, for other persons eyes too, so to speak. You are not alone when you make decisions about how to write, and things like that. Early in a project so, if it actually is a job for the requirements engineer can be questioned, but the requirements engineer together with the subproject leader and some others, to divide the project into pieces. In which order should we write the requirements? In which order should we implement the requirement? How do we divide the requirements in the different subsystems, for example?” (Requirements engineer)

Quotation 25: “We have decided upon which use cases that needed. Which relationships there should be between them? [...] Decide which actors there are. Between the subsystems, you call the other subsystems actors.” (Requirements engineer)
Quotation 26: “Then there is also on what level the requirements should be. That also demands a lot of experience. When you talk about use cases, you usually say that they can be on the bottom level and the surface level and up in the sky.” (Requirements engineer)

Quotation 27: “Well, you have a constant decision. It is the level of details. What should I write as a requirement and what should I not write as a requirement. […] It is a constant judgement. What do you include? What do you not include? How detailed should you be? This also then, all the beautiful empty rhetoric that all such requirements lecturers who talk about that you should describe the borders of the system with your requirements and not inside the system. […] But this is a great guiding principle, so to speak. However, when you are really sitting and writing something, then some requirements are not possible to write without some knowledge about how it works inside. […] It is easy to say at such requirements lectures, that you should put requirements on the border and not how the system looks inside. However, you don’t land there in reality. You always have such small decisions all the time, so to speak. What should I include? What should I not include? Is this construction? How closely do I describe it?” (Requirements engineer)

Quotation 28: “Well, which requirements are essential on the whole for the main functions? Which requirements are essential now? Have some understanding of the order of priority, when to do what. And it is also a priority how deep I should go. Because you cannot go equally deep and think about… for every requirement. You have to get a feeling for what should I spend my time on.” (System engineer manager)

Quotation 29: “It is this that is the difficulty, to have this balance. When should we do deep investigations and when should we not do them. It is this that is the difficulty. […] It is difficult to know when to do a lot of work and when you should chance a bit. Because you have to chance a bit.” (Focus group participant)

Quotation 30: “The first, the most important judgement, this we therefore do from the very beginning. It is to set some limit for how far you should investigate something. What level of ambition you should have and this is just a judgement that you just… It becomes easier and easier with experience. […] Good enough, this judgement I personally find very difficult.” (Requirements engineer)

There is also an authorisation routine in the decision process model of Mintzberg et al. (1976), in which the decision is approved by someone in order to commit the organisation to a certain course of action. In the empirical data, we have not found any specific authorising decision-making activities. Nevertheless, there can be authorisation activities in the case study company.

The implementation phase

Decisions trigger actions, and in the implementation phase several decision activities are performed, such as communicating decisions, plan actions, and track performance (Power, 2002). In this phase, the RE decision-makers set up the requirements documents to be used (see Quotation 31). They document trade-offs, decisions and rationale for decisions together with the functionality (see Quotation
They check design specifications (see Quotation 33), and support the persons, who verify, design, and implement, to interpret the requirements (see Quotation 34). The RE decision-makers are a service function for all requirements stakeholders (see Quotation 34 and Quotation 35). The decision communication carried out is dissemination, i.e., the RE decision-makers inform others of the decisions.

Quotation 31: “Typically, a requirements engineer can do that then [i.e. technical investigations] and you can document that in different forms of reports; short reports or long ones, e-mails or oral presentations or sometimes in requirements. But it does not always result in requirements.” (System engineer manager)

Quotation 32: “Then I sat down and made small reports of how the function should behave for the different functionalities. […] Which trade-offs have we made? Why did we make this decision? Things like that are also important to document, because you may see things when you are doing analysis that is very important. Okay, we have to take this knock that we make this half good because otherwise it will not work at all. You must document this, since in six months you have forgotten why you had to make that bad decision. Then you might say, oh we will change this to the better. And then it perhaps does not work. You have forgotten the important details.” (Requirements engineer)

Quotation 33: “Then you are also making checks. You are called to all the checks of design specification as a requirements engineer.” (Requirements engineer)

Quotation 34: “One thing that you have to do a lot is to help the verifiers to interpret the requirements so that it will be correct. It is the same with the software engineers, who need some interpretation help. One does not always write the world’s best requirements that are unambiguous and easy to understand.” (Requirements engineer)

Quotation 35: “… both the requirements work and the requirements engineers are a service function in some way in the project, I think. All the time… you have to stay alert of what is happening around you and answering questions and…” (Requirements engineer)

Quotation 36: “The work is not completed when you have set a requirements specification. Instead, then it continues with constantly answering questions. […] Sometimes it is just with words, oral or via e-mail, explaining what was meant. So this is much a part of the role of the requirements engineer as writing the requirements themselves. The job is not finished when the document is ready. It is then it begins.” (System engineer manager)

The follow-up and assessment phase

In the last phase the consequences of decisions are checked. This may lead to the identification of new problems (Power, 2002). In this phase the RE decision-makers check the verification and test specifications (see Quotation 37). They also have user group meetings in order to validate the outcome with the users (see Quotation 38).
Quotation 37: “And… but I read their test specifications, so to speak. I am there checking their test specifications, how they plan to test the requirements.” (Requirements engineer)

Quotation 38: “When it concerns the MMI [i.e. the man-machine interface] then you also validate with the customer or well… not with the customer, but the user actually.” (Requirements engineer)

7.1.2 Management of requirements changes

This decision process concerns the management of requirements changes during the lifetime of the project. Hence, it occurs frequently during the project.

Like the other decision process, management of requirements changes is highly iterative and each instance of the decision process can take any path through the routines.

The identification phase

There are three different ways RE decision-makers recognise problems that initiate the decision process called management of requirements changes. There can be error reports from verification or construction, or there can be direct requirement change proposals that start the process (see Quotation 39 - Quotation 40). Requirements errors can also be discovered by the RE decision-maker and in such cases he or she carries out dissemination activities by writing an error report (see Quotation 41).

Quotation 39: “A requirements change can arise if there are changing requirements from the customer, but it can also come from someone in the organisation who discovers that this requirement is actually not good. Perhaps it is not adjusted in the best way to the equipment we currently deliver. […] And when there are requirements that are in fact correctly formulated, but written in a way that can not be verified or that will be too expensive to verify.” (System engineer manager)

Quotation 40: “You can discover it by yourself as a requirements engineer or you can receive it from the project management or system management that you are heading in the wrong direction, or are you really in control of this. So, who initiates or what initiates it… well, it can be urgent problems in the system or construction work, or something planned.” (Subsystem manager)

Quotation 41: “I write the error report when I, in relation to my work, see that there are errors in the requirements. But the largest number of error reports come from verification and construction, when they see that this is not possible to implement in this way or that the verification says that this requirement is not verifiable. Or the verification can discover that this requirement does fit at all. It is then meaningless to have an internal requirement that actually shouldn’t be there either.” (Requirements engineer)

In the diagnosis routine, the RE decision-makers carry out investigations (see Quotation 42 - Quotation 44). They check change proposals (see Quotation 45),
investigate error reports (see Quotation 47 and Quotation 48), and initiate themselves into input from the customer, depending on what initiated the decision process.

Quotation 42: “... in an investigation all possible things can be included, such as spreadsheets in Excel and advanced mathematical models. You try to model as well as possible how this will be in reality when it is implemented, so we often make such very advanced models and simulation tools.” (System engineer manager)

Quotation 43: “Will this affect, so to speak, how the system behaves externally? How will this affect what the customer experiences? As a rule, it is the first question you put to yourself. Do we need this requirement at all or can the software engineers do as they want?” (Requirements engineer)

Quotation 44: [Characteristics of a good investigation:] “Well, it is that there is a background and that it is exhaustive and that there is a background and the cause to why you want to do this. That it is clarified and explained what they want... you to do. The thing that perhaps, what you can think is most difficult to do; it is in the end a cost estimation. What does it cost to implement the whole?” (System engineer manager)

Quotation 45: “Well, it depends on what proposal it is. Sometimes such a proposal arrived with an internal investigation, that you have decided in advance to do an investigation and then you make a check in that case. [...] The reasons for doing it, profitability. Who will be concerned? When will it be implemented and...” (System engineer manager)

Quotation 46: “Most of the problem reports on requirements come from verification when they sit and make their verification specifications and when they are testing, so to speak. Then, where its actual origin is... Many times you end up in discussions that you have to discuss with the customer. That you have found a white corner in the system, or what you should call it, which actually is not specified how it should behave.” (Requirement engineer)

Quotation 47: “It is quite interesting that when it goes wrong, when it does not work in the verification. When something happens there, then as a rule... Well, then you easily make an error report on the requirement.” (Requirements engineer)

Quotation 48: “Often, when we receive error reports, we have a lot of input from the customer to go through.” (Requirements engineer)

The development phase
In the design routine, the RE decision-makers solve error reports (see Quotation 49), as well as change and add requirements (see Quotation 50). We have not identified specific activities that can be categorised either in the search routine or in the screen routine.

Quotation 49: “... you receive some problem and then in some form of forum... either with e-mail or that you meet and discuss... You try to find out how the
system should behave and how you easiest solve it in order to get there.” (Requirements engineer)

Quotation 50: “… it can, for example, be a missing requirement. It ought to be a requirement of something and this is noted almost as an incorrect requirement. That you make a proposal on a new requirement and then, at the next check occasion, you bring it up for judgment and add it or change it or whatever you do. You decide how to manage it any way.” (System engineer manager)

The selection phase

In the analysis mode in the evaluation-choice routine, the RE decision-makers check so that a change proposal is not going to become a problem for other subsystems (see Quotation 51 and Quotation 52).

Quotation 51: “I mean, the system must be connected, so that to the degree it affects other requirements you have to try bring in that too, so to speak. We don’t sit and tick it off against all other requirements, but… Instead, it is more concerned with our knowledge in the head, so to speak. That you know, okay, but then this will not be good and how is this related to that and…” (Requirements engineer)

Quotation 52: “Thanks to that we have this splendid [requirements management tool] and database system and that, as I said earlier, you become aware of when you are making some clear, nice little change that is like this… This is actually a requirement atom that is in seven other different projects and in some way an approval must be gained from all these seven projects and all these seven project has to discuss whether or not they will apply this change or if they will not apply this change, and if they can afford to implement this change, in which phase they are in…” (Requirements engineer)

In the bargaining mode, there is one system-related decision matter. This decision matter is negotiated when the requirement in question is shared with other projects:

• Is a requirement change proposal going to be approved or not?

There are also two work-related decision matters that are managed in this mode (see Quotation 53):

• When is the requirement change going to be activated?
• When is the requirement change going to be implemented?

In the judgement mode, there is one system-related decision matter, which is dealt with by the individual RE decision-maker when the requirement in question is project unique.

• Is a requirement change proposal going to be approved or not?

There is also one work-related decision matter to handle:

• How should the particular requirements change be managed?
Quotation 53: “...if it is some pretty large correction then you must decide when we should implement this correction, in which development step? Should we do the change at once or should we apply it later on? This is a pretty large decision that you have to make.” (Requirements engineer)

In the authorisation routine, we have identified activities. When there are particular important decisions, authorisation is needed concerning requirements changes (see Quotation 54).

Quotation 54: “It can be an unfortunate language blunder that makes the requirement difficult to understand, well, then you just update and correct it. On the other hand, it can be something that represents a large reconstruction. Then we have to raise ourselves to a suitable level before it is decided.” (System engineer manager)

The implementation phase

In this phase, the RE decision-makers generate requirements documents (see Quotation 55 and Quotation 56) and documents aimed for the verifiers and implementers that show the differences between former and current requirements documents (see Quotation 57).

Quotation 55: [Concerning accepted requirements change proposals:] “Well, it goes into the specification that is revised and then it is disseminated to the concerned subsystems.” (System engineer manager)

Quotation 56: [Concerning rejected requirements change proposals:] “Well, then it will be a notation. It is just noted, either in the check record or already in the error report maybe.” (System engineer manager)

Quotation 57: “Then I also generate some delta documents to my verifiers and software engineers that precisely tell what has happened between different revisions and things like that.” (Requirements engineer)

The follow-up and assessment phase

We have not identified any specific activities in the decision process that can be categorised as belonging to the follow-up and assessment phase.

There are similarities between the two decision processes. More important, though, are the differences between them.

7.1.3 Differences between the decision processes

In the case study we found two different RE decision processes: a) establishment of requirements and b) management of requirements changes. Although they have similarities, there are also important differences. Both processes are highly iterative and include several sub-decisions. The first decision process occurs once in the lifetime of a project, while the second decision process occurs frequently. The first process embraces more decision-making activities than the second one. Some of the
decision-making activities are shared by the two processes, and some activities are unique to each process.

The main point in separating the two RE decision processes is that it makes it possible for us to examine the diversities and identify dimensions in the RE decision-making activities. This would have been more difficult if we would have tried to describe one generic RE decision process. These differences imply more RE decision support characteristics, which are elaborated in chapter 9. The main differences between the two RE decision processes are summarised in Table 1.

<table>
<thead>
<tr>
<th>Table 1, Comparison between the two RE decision processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Establishment of requirements</td>
</tr>
<tr>
<td>Decision recognition</td>
</tr>
<tr>
<td>Diagnosis</td>
</tr>
<tr>
<td>Search and screen</td>
</tr>
<tr>
<td>Design</td>
</tr>
<tr>
<td>Analysis</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Evaluation/choice – bargaining mode</td>
</tr>
<tr>
<td>Evaluation/choice – judgement mode</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Authorisation</td>
</tr>
<tr>
<td>Implementation</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Follow-up and assessment</td>
</tr>
</tbody>
</table>

The decision recognition routine of establishment of requirements begins, in our case, with the arrival of a set of higher-level requirements. As a consequence, the main activity of RE decision-makers in the following routine, i.e. diagnosis, is
comprehension activities. These activities embrace gaining knowledge of the problem domain, which includes the system as a whole and the customer needs. Thus, these decision-making activities have a larger scope compared to the other decision process. The other process, management of requirements changes, starts with error reports and change proposals. Accordingly, the decision-making activities in the diagnosis routine are focused investigations. Hence, these activities have a more narrow scope than in the other decision process. Another difference is the amount of information. The amount delivered in the decision recognition routine is usually lesser in the management process compared to the establishment process.

In the search and screen routines of establishment of requirements, the RE decision-maker search for existing components to reuse. In the other decision process, management of requirements changes, we have not found any decision-making activities in the empirical data that can be categorised in these routines. Nevertheless, there may be such activities, for example, searching for solutions of similar error reports, although not present in the data.

The design routine involves different challenges for the RE decision-makers depending on decision process. In the establishment process, the activities concern creating use cases and requirements. Thus, there are creative challenges and the effort is more about idea generation compared to the other process. In the management process, RE decision-makers’ tasks are solving errors as well as changing and adding requirements. Therefore, the main activities, and thus the main challenges, are problem solving and idea evaluation.

In the analysis routine of the establishment process the RE decision-maker conducts requirements checking, requirements tracing, and risk analysis. In the management process the analysis concerns the change proposal in order to state its potential consequences. In the first mentioned process, the RE decision-makers work in teams with several other actors. In the other process, they, to a larger extent, carry out the analytical activities on their own. However, they communicate with other stakeholders, but stakeholders do not participate that actively in these decision-making activities.

In the implementation phase of the establishment process the RE decision-makers take an active part. They document, check design specification, and last but not least act as a service point for all requirements stakeholders. In the management process, the role of the RE decision-makers is comparatively of less importance. They are not, in this phase of the process, engaged in that many decision-making activities.

Thus, some of the decision-making activities are similar in the two processes, and some activities differ. However, both processes have several decision matters and decision outcomes, which have certain characteristics.
7.2 Characteristics of decision matters

The decision matters that are dealt with by RE decision-makers have certain characteristics, which are summarised in Figure 26. In Figure 11 on page 45, which is the origin of Figure 26, there was a fifth characterising dimension termed ‘focus with regard to domain-specific functional areas’. We did not find that dimension applicable in this case.

![Figure 26, Characteristics of decision matters](image)

The degree of structuredness is semi-structured. This means that the decision matter is partially structured and partially unstructured. For example, when deciding the behaviour of the system, some system behaviours are more or less necessary due to certain well known circumstances. Hence, such parts of the decision are structured. Other system behaviours are more difficult to decide. Perhaps they have to be generated, analysed and weighed against conflicting goals before a decision can be made.

The scope and time frame is that the decision matters are categorised as either tactical or operational. Tactical decisions affect a part of the organisation for a limited period of time. An example of a tactical decision matter is deciding in which order the requirements are to be implemented. Such a decision directly affects the work process of several actors within the current project. Operational decisions affect the project on a more daily basis. An example of such a decision is when an RE decision-maker decides what level of effort an investigation, e.g., of a function to write requirements shall have. The scope and time frame of a decision is then narrow. The decision matters handled by the RE decision-makers in the case study have not been found to be strategic.

There are general functions that can be related to decisions matters. The functions in the POCCC view are planning, organising, coordinating, commanding, and controlling (see section 3.1 on page 47). The decision matters in this case are related to planning, coordination, and commanding. These general functions are intertwined in the decision matters. Planning is outlining what to do, for example, deciding when a requirements change is going to be current. Coordinating is carried out when activities are harmonised. An example of this is when decisions are made concerning how requirements changes are to be managed. Commanding is when a process is
started, a process that is targeted towards reaching a certain goal. For instance, this is made when deciding if a requirements change proposal is going to be approved or not.

The degree of negotiations is that both unilateral and negotiated decisions are made. Unilateral decisions denote that several persons are involved, but there is one person who actually makes the decision. An example of a unilateral decision is deciding if the requirements can become baseline. In negotiated decisions multiple persons have to agree, such as deciding which system requirements belong to which subsystem.

The decision outcomes have not been discussed extensively enough in the interviews or in the focus group session. However, some inferences can be made. The system-related decisions have, in our case, more visible outcomes than the work-related decisions. When requirements have become baseline they are disseminated via a physical document and the requirements are placed in a certain tool. Requirements change proposals are marked current when that decision is made. Use cases, system actors, behaviour and appearance of the product are documented. The work-related decisions are not made clear to the same extent. Instead, such decisions usually only result in, for instance, a new or different way of working. The characteristics of the decision outcome probably have consequences for both system as well as the systems engineering process. However, this has not been elaborated enough for us to make any claims with respect to this.

Another aspect related to the decision-maker is the information sources that he or she can use during a decision process.

7.3 Information sources used by RE decision-makers

There are different types of information sources that RE decision-makers use in their decision-making activities. The types of sources that were identified in the case study were:

- Requirements and requirement-related information
- Customer
- Points of view
- Records and reports
- Theory
- The Internet

The source requirements and requirement-related information consists, in this case, of customer requirements, system requirements, and subsystem requirements, all of which concern the current system under development. There are existing requirements with an existing code from related systems as well as information about how product components used have previously been assembled. There are requirement change proposals, and a matrix concerning system requirements allocation between different subsystems.
The source customer delivers, for example, technical data of their existing systems, which are going to interact with the system to be. The customer is also a communication partner. The RE decision-maker obtains points of view from internal stakeholders, such as system manager, resource personnel, e.g., cognitive scientists, project manager, software engineers, as well as other RE decision-makers.

The source records and reports, for instance, consist of error reports, reports from investigations concerning functions, design reports, records from requirements check meetings. Informal records are, e.g., e-mail exchange with stakeholders. It also happens that the RE decision-makers use relevant theories, e.g., radar theory. They also use the Internet to find more information, for example, about concepts in the documentation that are difficult to understand.

The information varies along characterising dimensions. These dimensions can be seen in Figure 27.

![Figure 27, Characterising dimensions of information](image)

The information can be more or less governing or advising. Sometimes the RE decision-makers have to base their decisions on the information, e.g., higher level requirements, and sometimes they can use the information as a guide, e.g., point of views given by internal stakeholders. The information is given verbally or written, or a combination of these two. The information can be more or less formal. For example, there can be formal records or informal e-mail conversation. The information can come from internal sources, e.g., error reports, as well as external sources, e.g., technical data from the customer. Some information is created by the RE decision-maker him or herself, such as reports from investigations. Other information is created by other stakeholders, such as higher level requirements. Information along all dimensions may be of equal importance to the RE decision-maker.

### 7.4 Chapter summary and reflections

In this chapter, we, based on an empirical case study, describe two different RE decision processes, their decision-making activities and the decision matters they
encounter. We also elaborate the characteristics of the decision matters as well as the information sources used by RE decision-makers. We focus on requirements engineers, viewed as RE decision-makers, on a subsystem level in a contract development context, where projects are large and systems to be developed are highly advanced and complex.

To what extent is it possible to transfer the results to other organisations? It is not possible to fully answer this question. It is also not possible to paint a generalisable picture of RE decision situations, since every organisation is unique and every instance of a decision situation is unique. Nevertheless, to get an indication of the transferability of the case study results, we conducted transferability interviews. In these, we presented and discussed the findings with four experienced IT consultants in groups of two persons (see chapter 5 for details of methodological aspects). These discussions indicate that the results agree, to a large extent, with the experiences the IT consultants have of other types of organisations. However, there are also aspects that differ between companies. Thus, we can conclude that the results of the case study are partially transferable to other organisations.

All IT consultants thought there are two different RE decision processes, albeit there was some debate concerning aspects of this matter (see Quotation 58 - Quotation 60).

Quotation 58: “Yes, there are different processes. I think that. When you have baselined the requirements... before you have baselined the requirements, then you are concerned with establishment. But when you have baselined the requirements you enter the management of requirements changes.”

Quotation 59: “That coincides with how it looks today, as a snapshot of our reality.”

Quotation 60: [Interviewee A:] “Well, when a project begins... from a general point of view, there are two angles of this. The first one is... you have some sort of legacy from previous projects of a similar kind. That the project develops a new model of something or something like that and then you inherit requirements and then it is this part with creating new requirements that is for just this new thing, so to speak... It is two things I don’t... Establishing requirements... I don’t know if you should split this or if a point is missing there.” [Interviewee B responds:] “I think that, although there are requirements you already have, you have to collect them and make them current, independent where they come from. There are always some basic requirements and then there are some new ones and some that we don’t know of. And then you packet. It is a bit of packeting of the whole model.” (Discussion between two IT consultants)

The IT consultants agreed with our description of the identification phase for both RE decision processes and nothing particular was discussed. Concerning the development phase, the IT consultants had more remarks and did not agree to all aspects, e.g., regarding searching existing components to reuse (see Quotation 61 and Quotation 62) and that the differences between the two processes are not so clear (see Quotation 63). However, they agreed to some aspects (see Quotation 64).
Quotation 61: “My experience is that this reuse thing, especially code, is so to speak a buzz word. It... you can look for a long time until you find something that fits pretty well, but most often it ends anyway in writing something new. I seldom see that you have used a component in different projects. Though, often you can take ideas, but the code must most often be rewritten.”

Quotation 62: “Well, I thought for searching existing components to reuse, which feels like an important part of management of requirements changes, the other column there. [...] I don’t see it that clear, so to speak, as it says there in establishment of requirements, but... [...] I would like to switch them... or to my experience, it feels more like you work this way here [i.e. in management of requirements changes] than that you... Search and screen is later on in the project, so to speak, under some more... Not in an early phase, but so to speak…”

Quotation 63: “I have a feeling that you are often in a grey zone between these [...] so that it becomes a mix between this earlier establishment and management.”

Quotation 64: [Concerning the “empty box” of the search and screen routines of management of requirements changes:] “… that a change proposal arrives on it, often it is not large things, so that it is not worth throwing away what you already have made and search for something else new. So, from that point of view it agrees pretty well.”

Concerning the selection phase, the IT consultants had comments on the analysis routine. They thought that the analytical activities differ quite a lot depending on the line of business (see Quotation 65 - Quotation 67). Regarding the evaluation/choice routines as well as the lists of decision matters in the RE decision processes, they all agreed and they meant that these are decisions that are made (see Quotation 67 - Quotation 70).

Quotation 65: [Concerning the analysis routine:] “In my case it agrees well. The main focus is in this establishment…”

Quotation 66: “It is also very much about what kind of result that is the output of the project. I understand that [the company in the case study] is careful and performs risk analysis and things like that, although it is not apparent in management of requirements changes. [...] I know that medical technology is also very careful, even when it concerns management of changes, to do analysis and traces and everything. [...] some other lines of businesses don’t do things like that at all, just do the change. It is probably very dependent on the line of business... what the product looks like in the end.”

Quotation 67: “When it comes to changes, it is often an economical analysis that you do. Is this worth doing?”

Quotation 68: [Concerning evaluation/choice routine, judgement and bargaining modes:] “Well, there are advantages and disadvantages with this of course, but I think this is the way it is.”
Quotation 69: [Concerning the lists of decision matters:] “I recognise it. I try to figure out if something is missing or so, but I don’t.”

Quotation 70: [Concerning the lists of decision matters:] “All of these are questions that they put on themselves, clearly. However, what is interesting is if something is missing. I cannot think of anything right now.”

The IT consultants thought that the activities in the implementation phase and the follow-up and assessment phase were correct. However, they discussed that the feedback from the verification to the RE decision-makers, for example, is frequent and important (see Quotation 71).

Quotation 71: “It feels like it is in this phase… that you come with a whole bunch of error reports and other types of feedback so to speak. Documents in this case can be test reports and error reports and performance values and such things […] feedback, that relates to the requirements.”

Thus, the decision situations of RE decision-makers in the case study are not unique and they have transferability potentials. However, it is not our role to state the transferability. Instead, the judgement of transferability lies with the person who wishes to use the results (Lincoln & Guba, 1985).

In this chapter, we present parts of the decision situation of RE decision-makers. However, there are also factors in the decision situation that cause problems and difficulties for them.
8 Factors that affect the RE decision-makers

In this chapter, we present factors that affect the RE decision-makers. Six general factors in the case study that directly or indirectly have an effect on the RE decision-makers have been found. The factors are: attitudes towards requirements work, communication and coordination, resources, pressure, cognitive load, and knowledge. The study indicates that problems and difficulties related to these factors pose threats to the quality of RE decision-making. As seen in Figure 28, the factors are related to each other. It is reasonable to assume, however, that these factors are not a complete set of factors and more relationships than the ones found may exist. To quote Reisberg (2006, p 464) “any factor that influences our thinking in general (our ability to make judgement, our ability to reason) should have a direct impact on decision-making.”

![Diagram of factors affecting RE decision-making]

Figure 28, Factors that affect the decision-making of requirements engineers

Direct quotations are included in the text for illustration. They are highlighted with help of indentation and font size. All quotations are translated from Swedish.

8.1 Attitudes towards requirements work

A source of several problems in the RE decision processes is the status of the RE discipline and the attitudes of individuals in the systems engineering process towards requirements work. We have identified three types of problems:

- Low status of requirements work
• Prestige between subsystems
• Departmentalisation of work

RE as a discipline and as a competence has generally a low status. There is to some extent a lack of understanding of the difficulties that face a requirements engineer and the actual benefits of requirements work (see Quotation 72). The low status of requirements work has consequences. To become a requirements engineer is not viewed as a step upwards in someone’s career (see Quotation 73). Thus, experienced software engineers do not want to have that role. Instead, engineers fresh out of university are often recruited (see Quotation 74). Since there is a lack of understanding in the organisation of the difficulties of being a requirements engineer, education in RE is not, as far as we have seen in the case study, a requirement for becoming a requirements engineer. This results in requirements engineers having limited experience and limited knowledge (see Quotation 75) about the product as such, the domain in which the product is to be used, and the RE tasks (see Quotation 76). The low status and lack of understanding also have other consequences. There are fewer possibilities to improve qualifications and fewer and less developed support tools. A couple of interviewees compare the support given to requirements engineers and the much more developed support a compiler gives to a programmer.

Quotation 72: “There are persons where I work that seriously say that this is only blah-blah, we shouldn’t do this.” (Focus group participant)

Quotation 73: “It seems not that popular for software engineers to go to either requirements or verification. It should actually be required that you have worked a while and gained some experience before you can… Since it is difficult, so to speak, you need a network within the company in order to know which persons you should talk to in a project when it comes to coordinating different stuff. As a person fresh out of university you generally don’t know the people that work in the company.” (Subsystem test manager)

Quotation 74: “… it has not always been so attractive for the software engineers to go to that job. […] And then, we have recruited externally directly to this role.” (System engineer manager)

Quotation 75: “Many of those who have become requirements engineers have come to [the company] and they have not worked as an engineer before and they have not worked in the organisation before. And then you perhaps don’t have the ability to… have enough authority in your role and you can only gain this authority by showing that you know what you are doing, thus by knowledge. Then I also think that many of them [i.e. the requirements engineers] have always been half a step behind and then it is really hard to catch up and be half a step ahead instead.” (System engineer manager)

Quotation 76: [Interviewee A:] “But to go a course to learn about requirements, it doesn’t exist today.” [Interviewee B:] “No, and not how you should act in these complex situations then, so to speak, when they demand some things, create requirement on new things.” [Interviewee A:] “It is because they think that
requirements engineering isn’t anything to invest in.” [Interviewee B:] “It is not a competence.” (A discussion between focus group participants)

There is prestige between at least some subsystems (see Quotation 77). Each subsystem group wants their subsystem to be important on its own. They do not want their subsystem to be just a resource for other subsystems (see Quotation 78). This affects how requirements related to the interface between subsystems are written (see Quotation 79). The requirements tend to include more detailed solutions, rather than only describing the behaviour and characteristics of the system. There is also a risk that the solution for the system as a whole is not optimised due to this prestige (see Quotation 80). Thus, decision-making in RE is directly affected.

Quotation 77: “And don’t tell us too much, but just tell us exactly what we need to know. Don’t tell us more than we want to know. [...] everyone wants the “box” that is the border and they want help with some things, but concerning other things they don’t want anybody else to interfere.” (Focus group participant)

Quotation 78: [Interviewee A:] “Is there much prestige between groups?” [Interviewee B:] “Yes, but if you make changes in the organisation, it will be that every new organisation is responsible for their part. No one in the new organisation wants to be just a resource pool, instead you want to have responsibility of a subsystem.” (Discussion between focus group participants)

Quotation 79: “… in reality the subsystem mirrors our organisation – lines. There are, for example, places that almost only work with [X] and they work with [Y], they work with [Z]… This sometimes causes the interfaces to be unnecessarily complicated. You can easily think that an interface will be much simpler if you, e.g., moved the border between [subsystem X] and [subsystem Y] a bit in one or the other subsystem. […] This has an impact on our requirements specification. If you had that insight you would get much simpler communication.” (Focus group participant)

Quotation 80: “There are some, we talk subsystems now, which attract more problems, because there are subsystems that are unwilling to make changes. Even if something would fit better into another subsystem, it is often this more reasonable subsystem that takes on the responsibility to do this.” (Focus group participant)

There is also departmentalisation of work between different groups within the development process (see Quotation 81). This is shown by the fact that information is sometimes not forwarded to requirements engineers (see Quotation 82). This directly affects the possibilities of making relevant and reasonable decisions about requirement.

Quotation 81: “… these territories again, you aren’t allowed to talk about things in that subsystem for example. Then they feel attacked in some way for some reason.” (Software engineer)

Quotation 82: “You cannot participate from the customer phase, but at least in the system phase, in the analysis and everything. How you divide? Why the
Both prestige and departmentalisation of work can be further analysed and related to the research domain of organisational decision-making. Organisational decision-making can be viewed as a political process (Klein & Methlie, 1990). A central concern is power, which can be seen as the potential to influence people how they act and think. Power is important both in the activities preceding a decision and in the implementation of a decision (Pfeffer, 1992). The amount of power involved in a decision process depends on the characteristics of a decision. Power is used to a greater extent in situations such as interdepartmental coordination, and used to a lower extent in situations such as work appraisals (Pfeffer, 1992). In the political view of organisational decision-making, information is viewed as a resource that can give power depending on how it is used. The power of individuals and roles within the organisation is also viewed as an important aspect (Browne, 1993).

8.2 Communication and coordination

Decision communication is carried out in the whole decision process (Mintzberg et al., 1976), i.e., both the decision matter and the decision outcome can be communicated. The factor communication and coordination was ranked as the factor with highest priority by the participants in the focus group. It consists of four problems:

- Lack of coordination of way of working
- Little involvement in discussions
- Time-consuming coordination (with respect to calendar time)
- Little communication of decisions

The way of working with requirements is not coordinated enough. This entails requirements not being written in a consistent way. Each requirements engineer has his or her own way of writing requirements (see Quotation 83), so there are different styles, different levels of detail, and there is a lack of cohesion among the requirements (see Quotation 84). The consequence of this is that the requirements specification becomes muddled and more difficult to use (see Quotation 85). This affects the cognitive load both for the requirements engineers themselves, but also other actors that use the requirements in their work.

Quotation 83: “In principle, I can look at any use case and see which person has written it. So it is like that... We have so different ways of writing it actually.” (Requirements engineer)

Quotation 84: “However, our problem is that we have placed ourselves at a detailed level in some cases and in other cases the requirements are too abstract. There is no single way to write actually.” (Requirements engineer)
Quotation 85: “The requirements are made very messy. It is several different persons who have written the requirements. It feels like that there is no uniting trend, main thread in the whole. Everyone has had their own style.” (Requirements engineer)

The requirements engineers have too little involvement in discussions, especially in discussions in early development phases. Since they do not take enough part in discussions with customers and in those concerning systems requirements (see Quotation 86), it becomes more difficult to understand the problem to be solved by the system and the system as a whole (see Quotation 87), i.e. this increases the cognitive load. Since requirements engineers are often inexperienced, the actors at the system level and the software engineers avoid forwarding information to requirements engineers. The system level actors and the software engineers are afraid that the requirements engineers unintentionally distort the information (see Quotation 88). This can directly and negatively affect the RE decision-making, since the requirements engineers may not have enough or correct information to base their decisions on.

Quotation 86: “A lack here is then that we requirements engineers in the subsystems did not take part in the earlier contacts with the customer, when the customer requirements were specified. The first would be to steer it so that we can get a bit more clearly with regard to the requirements and secondly to understand what the requirements actually meant and what the requirements refer to. […] Assumptions that we don’t know of, that cause our interpretation to be something that the customer doesn’t agree to.” (Requirements engineer)

Quotation 87: “To get an overview of what is going to be implemented, it would be great to take part from the beginning.” (Requirements engineer)

Quotation 88: “Well, this means that the requirements engineers will always be a step behind or a half step behind. […] is why people do so both from construction and from system management, it is that you want to make sure that you have understood it. You don’t want to go a detour via something [i.e. requirements engineers] that you are afraid will distort the information.” (System engineer manager)

Requirements changes may concern other subsystems or the system as a whole. Some requirements are used by two or more projects at the same time. This causes a need to coordinate requirements changes within the project and between different projects. The coordination can be time-consuming, especially with respect to calendar time (see Quotation 89 and Quotation 90). This increases the pressure on the requirements engineer.

Quotation 89: “Well, actually you ask how much time it takes to process our problem reports and is it such things that currently don’t work, so to speak […] I mean, it can take a half year before everyone has said what they want to say and changed their minds ten times and it can take a very long time before our problem reports are put straight. It is not because it involves a thousand hours of effective work. Instead, there are very few hours of effective work, so to speak. However, it takes time and there are a bunch of people that have a bunch of
opinions in the beginning and then, after a while, there is some project manager that says, well, we don’t have money for this. We will not change this. It can take a long time.” (Requirements engineer)

Quotation 90: “Well, this can be discussed endlessly and without ending up somewhere, so to speak. They don’t really put down their foot and when it becomes… […] It takes a long time and system probably talks to the customer and there can be endless discussions that are very long-winded and in the mean time we construct.” (Requirements engineer)

Once a decision is made, there is a commitment to a certain course of action or a commitment to passiveness. The next step in the decision process is implementation of decision (Power, 2002). In decision implementation decisions are communicated, actions are planned, and performance should be monitored (Mintzberg et al., 1976), (Power, 2002). In the case study company, there is a problem that decisions are not always made clear to the persons that are supposed to act upon them (see Quotation 91). For example, the outcome from discussions concerning decisions is not always documented (see Quotation 92). Another example is that software engineers overlook certain requirements (see Quotation 93), e.g., performance requirements (see Quotation 94), because their attention is not drawn to them when needed (see Quotation 95 and Quotation 96). Thus, the RE decision process is negatively affected.

Quotation 91: “We make decisions, we don’t document them and we disseminate them to the wrong people. We can be much better there.” (Focus group participant)

Quotation 92: “There we have a huge problem that we don’t have the documentation of why it became as it became or why we did as we did.” (Focus group participant)

Quotation 93: “… some details float around in the organisation and are really important. We forget large technical problems that are like ticking bombs. We should have dealt with them immediately.” (System engineer manager)

Quotation 94: “Performance requirements are actually rather critical; however they don’t feel that way, unfortunately. […] We did not think about optimising the performance when we made our construction. It just became… We only prioritised the functionality. It is not critical to write it then, to put it in the [requirements specification], but instead draw peoples attention to the requirements.” (Requirements engineer)

Quotation 95: “Then, we cannot absorb information, but… I wasn’t even aware of the existence of this protocol until it was time to do the implementation. […] Someone should have had, should have felt it their responsibility to convey the information about this…” (Requirements engineer)

Quotation 96: “We have a whole bunch of requirements that nobody thinks about until we are going to verify them […] It should have been more critical maybe. To inform people that there actually are requirements.” (Requirements engineer)
8.3 Resource

There are three problems related to the factor resource that affect the decision-making of a requirements engineer. These are:

- Usability problems in requirements management tools
- Lack of external expertise
- Lack of introduction to and education in RE

The requirements management tool used in the case study suffers from usability problems, such as effectiveness, efficiency, as well as learnability. The tool is described by the requirements engineers as cumbersome to use, and there is a high learning threshold (see Quotation 97). The actors related to the requirements engineers claim that the usage of the tool has caused illegible requirements (see Quotation 98). Thus, the effectiveness of the tool is low (see Quotation 99). As a result of these problems, the use of requirements management tools has had low penetration in the organisation (see Quotation 100). Only the requirements engineers use the tool, while, for example, the actors at the system level or the software engineers do not. The requirements engineers use the tool despite its usability problems because it has utility qualities, i.e., certain useful functionality (see Quotation 101). Thus, the inadequate use of requirements management tools increases the cognitive load of the requirements engineer, and hence negatively and indirectly affects the RE decision process.

Quotation 97: “... this [requirements management tool], it is a high threshold to start working with it. Many people don’t like that. It is a matter of experience actually, partly to start thinking... to think in some structured way of thinking. You aren’t used to that either. So there is a large threshold to... with this. It is a bit troubling. However, it is actually so that the tool doesn’t really... You are used to clicking and things like that, so to speak, and always get a lot of feedback of where you are and what you are doing. Such things aren’t afforded by this tool, instead you have your little prompter and your small boxes to write in. You have difficulty to see the whole picture, so to speak.” (Software engineer)

Quotation 98: “... how the requirements tool was used three or four years ago... and this led to documents that, to us who didn’t write them, were totally illegible.” (System engineer manager)

Quotation 99: “There are a lot of tools for requirements engineering and there is nothing wrong in that. However, one gets a lot of aid to do many errors quickly if one uses a tool in a wrong way.” (System engineer manager)

Quotation 100: “Because an intuitive and easy-to-use tool that... I think it would have a quicker penetration in an organisation, than if you have a cumbersome tool that is difficult to use [...] which purposely trips you up so that you get really tired. [...] However, the very basis of having requirements management in database form, that I think is completely obvious.” (System engineer manager)
Quotation 101: “Well, I don’t experience it as very intuitive or so, even if it of course has a lot of powerful functions. I would like it to be intuitive on a level such as in Access.” (System engineer manager)

_External expertise_ is sometimes needed. For example, requirements engineers who work with the user interface may need to ask cognitive scientists for help. In the case study, there are too few cognitive scientists available, which sometimes forces the requirements engineer to follow his/her gut feelings (see Quotation 102). This directly affects the RE decision process in a negative way.

Quotation 102: “So that I feel, I go very much on feeling. I have to do that. Check with her [the cognitive scientist] if it is up the creek.” (Requirements engineer)

There is not enough _introduction to and education in RE_ for novice requirements engineers, which makes it difficult to carry out the RE tasks (see Quotation 103 and Quotation 104). This affects the factor knowledge-related problems, and thus indirectly affects the quality of RE decision-making in a negative way.

Quotation 103: “… what I think is missing is to take care of… acclimatisation of the tasks.” (Requirements engineer)

Quotation 104: “It is always a problem when people are fresh out of university. They need time to be trained. You probably often underestimate that. It takes… to become very skilled in your discipline it takes several years before you have become experienced. Then, there is also a high turnover of staff.” (Subsystem test manager)

### 8.4 Pressure

Two problems in the factor pressure are identified:

- Time pressure
- Several actors with different needs

There is often _time pressure_ involved in the RE process. The requirements engineers are given the needed information too late, in their opinion, so that there is not enough time to work thoroughly with the functionality and to investigate how different functions affect each other (see Quotation 105). Because the software engineers also have time stress they begin to construct before the requirements are written (see Quotation 106) basing their work on hearsay instead (see Quotation 107). This results in incorrect code, which becomes clear when the requirements are written. An additional problem is that the software engineers then experience that the requirements are an obstacle in their work and not an aid (see Quotation 108). Time stress is often found to be a part of a naturalistic decision-making setting, which causes pressure on the decision-maker. With severe time pressure decision-makers tend to focus on a subset of the available information and less complicated decision strategies may be used. When subjected to severe time pressure decision-makers rely on heuristics to a higher extent (Payne & Bettman, 1988).
Quotation 105: “Their basic problem is probably that they [i.e. the requirements engineers] will always end up this intermediate position. That they receive basic information from the system level too late and that the construction must start early. So, often this time span that they need to do their work becomes difficult to…” (System engineer manager)

Quotation 106: “… finally the time runs away. The deadline when everything has to be delivered is approaching. Then the construction has to start prematurely. […] in this unit, there are happy software engineers working on hearsay. […] That one say how it should work and someone builds based on that.” (Subsystem manager and software engineer, i.e. one person with two roles)

Quotation 107: “We [i.e. the software engineers] are probably very sloppy with that [i.e. using the requirements] actually. It is probably because… so to speak… I can just tell how it works here. Here we have worked very much on hearsay.” (Subsystem manager and software engineer, i.e. one person with two roles)

Quotation 108: “However, it is always frustrating for those who write the requirements as well as for those who carry out the verification, since as a software engineer it is highly beneficial if the requirements are wrong. […] They aren’t seen as something that facilitate the work, but an obstacle that… that messes up things.” (Subsystem manager and software engineer, i.e. one person with two roles)

There are several actors that use the requirements, and they have different needs. Different actors want different levels of detail, and they want requirements to be written in a certain way (see Quotation 109). This problem is also discussed in naturalistic decision-making theories. When there are multiple players involved there is a risk that the players do not share the same understanding of goals and situational status. This can entail that the information needed in the decision process is not brought forward (Orasanu & Connolly, 1993). Since there are multiple players involved there can be shifting, ill-defined, or competing goals. In such cases, there are several purposes that direct the decision-maker and some of the purposes may not even be clear (Orasanu & Connolly, 1993).

Quotation 109: “… software engineers always think that there are too few details and are those from system think that there are too many details.” (Requirements engineer)

8.5 Cognitive load

Cognitive load is a “construct representing the load that performing a particular task imposes on the cognitive system” (Sweller et al., 1998, p 266). To put it in other words, cognitive load means a mental exertion. Its purpose is to interpret and process information in order to decide an action within a given space of time (Gulliksen & Göransson, 2002). Our mental capacity to consciously process information, which is called controlled processing, is limited. Thus, it is important that the individual uses this limited capacity for the most important and mentally
demanding tasks (Schneider, 1993). The cognitive load, in the case study, is expressed in three problems:

- Lack of general overview
- Lack of understanding
- High memory load

The large number of requirements and the limitation of the requirements management tools make it difficult to obtain a general view (see Quotation 110). An overview facilitates the understanding of a subsystem, its internal characteristics, as well as relationships and dependencies within and outside the subsystem (see Quotation 111). This problem has a direct affect on the RE decision process.

Quotation 110: “You see a lot of trees, but you cannot see the forest.” (System engineer manager)

Quotation 111: “They complain that they cannot see how things are related to each other. No, damn, it is not strange since there is no document that specifies how it is tied together. Instead, each person specifies his/her own function.” (Requirements engineer)

It is sometimes difficult to understand the meaning of a requirement. This is, for example, caused by ambiguity and bad translation. In this case requirements are written and read in English by Swedish peoples, and it is often more difficult to write well and obtain a correct understanding when a foreign language is used (see Quotation 112). Since existing requirements are used in decisions concerning requirements, and also other types of decisions, this directly affects the quality of RE decision-making.

Quotation 112: “This is a bit silly, since the customer requirements are in Swedish and then they are translated to English. Personally, I think this is unfortunate when we all speak Swedish. […] Hence, first translate… Someone translates it and then I have to translate it back to understand it. Then, I may not have interpreted it in the same way as the person who wrote it.” (Subsystem manager and software engineer, i.e. one person with two roles)

There is also a problem with high memory load. The requirements engineers have to keep several aspects concerning requirements in their minds. Aspects that are kept in the human memory are, for instance, dependencies between requirements and aspects in relation to requirements to look for (see Quotation 113 - Quotation 115). Thus, the RE decision-making is directly affected by this problem.

Quotation 113: [Interviewee A:] “… when you sit and work with requirements, I can miss it that you cannot see the dependencies between requirements. It can actually be that if you do a requirements change here, it doesn’t seem so very remarkable to change this thing. But if you do, it can be some other stuff that didn’t match that you aren’t aware of until you have come so far as… It may even have passed construction. [… ]” [Interviewee B:] “Well, it feels like they require a lot of space in the head, all the requirements.” [Interviewee A:] “You have to keep much in the head and it is not possible to write everything down.”
[Interviewee B:] “It is very hard. There is no other way than thinking about how can this affect the others. You have to remember well.” (Discussion between two requirements engineers in a group interview)

Quotation 114: “Then the decision part, then it is this that I mentioned before that you have to have all dependencies in you head. This is a decision which I find difficult. To decide upon requirements changes.” (Requirements engineer)

Quotation 115: “Well, you cannot write everything in the requirements. […] Don’t forget to check this and think about… stay alert and watch this… Because it isn’t possible, so to speak… There is nowhere else.” (Requirements engineer)

These problems can be visualised through a model of human information processing, as seen in Figure 29.

This model shows that we first encode environmental information to an internal representation, which is then compared to other representations stored in the human memory. In the third stage, we decide how to respond to the input and in the forth stage the response is organised and carried out. All four stages have an effect on and are affected by our attention, and at all stages our memory is used (Barber, 1988). If there are difficulties in one or more of these stages of our information processing, there is a risk that the decision-making is affected. An example of this is cognitive tunnel vision. This means that when we make a decision we have difficulties fully taking into consideration information that is not directly in front of us, even if we know that it exists and is important (Sandblad et al., 1991).

8.6 Knowledge

The low status of requirements work often results in a high turnover of requirements engineers. Consequently, there are sometimes knowledge-related problems among them. These knowledge problems concern:

- The domain (see Quotation 116)
- The product (see Quotation 117)
- Requirements engineering (see Quotation 118)
The knowledge-related problems influence both problem solving and decision-making. Problem solving is a mental activity closely related to decision-making, especially situations where the decision is unstructured and complex. In such cases, the solution to a problem is not obvious and work concerning identifying a problem and finding or developing alternate solutions for it has to be done.

Quotation 116: “Then the difficulty of the requirements work is that… It is that you must have the domain knowledge. It is… Before you’ve got a certain level of domain knowledge, then it is a hard time. Then it is really difficult. However, once you reach a certain level, it becomes much easier. You’ve got much more to relate your facts to.” (Requirements engineer)

Quotation 117: “We were rather inexperienced in the field. Then you have to trust in many other people. It was rather tough for a while, but you learn more and more so that… It is a very good way to learn a lot. Although, I think there are many that find this tough also. You have to enjoy learning things.” (Requirements engineer)

Quotation 118: “Like me, for example, that started to work with requirements. I would have needed formal education in; What are requirements? At what level should I land on? That… that I believe is rather important, that you know what a requirement is. How should you avoid being affected by others to write requirements that shouldn’t be written, that shouldn’t need to be written? How can I know that my requirement isn’t too fuzzy, so it could be misinterpreted and actually not be valuable? And that my requirements aren’t too detailed so I steer the construction in a direction that isn’t optimal. An important thing is not to write requirements that aren’t possible to verify.” (Requirements engineer)

There are differences between experts and novices in their ways of solving problems, since experts have gained experiences and skills that can make them more successful and efficient (Klein & Methlie, 1990; Zachary & Ryder, 1997). Experts are more efficient and can use their knowledge to draw correct conclusions, even if the available information is not complete. An expert does not start solving a problem through specifying all possible causes. Instead, experts quickly distinguish relevant information from irrelevant and possible causes from impossible ones. An expert can recognise the pattern of a certain type of problem, and connect a possible solution to it. Expertise consists mainly of domain knowledge, rather than general problem-solving strategies and methods. An expert knows the interrelationships between concepts, causal relationships. While the domain knowledge of a novice primarily consists of facts and basic concepts. The theoretical knowledge of an expert is both conceptual and analytical. In addition, an expert’s experiential knowledge is gained through training and practice. An expert has the ability to chunk sub-goals and have a global focus. The novice has a more local focus and treats sub-goals in a more sequential way. Experts have the possibility of performing case-based and more intuitive problem solving, and using strong domain dependent methods. Novices tend to be more analytical in their problem solving and they use weak general methods (Klein & Methlie, 1990; Zachary & Ryder, 1997).
The experience level also shapes the way decision-makers use information. It might seem apparent that the more experienced decision-makers have more possibilities to effectively use available information. However, there can be both positive and negative consequences. An experienced decision-maker may, for instance, have an increased potential to detect errors in a familiar set of data, but there is also a risk that he or she relies too much on a feel for the data (Fischer & Kingma, 2001).

Thus, the quality of RE decision-making is directly affected by the level of expertise the requirements engineer has.

8.7 Chapter summary and reflections

In this chapter, we, based on an empirical case study, describe a number of factors that directly or indirectly influence RE decision-making and as a consequence may affect the decision outcome. The factors are attitudes towards requirement work; communication and coordination; resource; pressure; cognitive load; and knowledge. We identify difficulties and problems related to each factor that can cause potential quality problems in RE decision-making. However, we do not look into the actual effects within the organisation, but instead motivate the arguments by using decision-making theories.

Although, there are also several other problems and difficulties in the RE decision processes, we have not been able to group these aspects into a named factor. Examples of such problems include:

- Missing requirements
- Domino effects
- Avoiding solution requirements
- Obtaining an accurate flow in broad use cases

When there are missing requirements there is lack of information. The interviewees also find it difficult to predict domino effects when new functions are implemented. They also sometimes find it difficult to avoid writing requirements that include solutions, and also to create accurate flows in broad use cases. All of these difficulties generate direct effects in the RE decision process.

As in the previous chapter, we can pose the question concerning to what extent it is possible to transfer this result to other organisations. As argued before, it is not possible to fully answer this question or make the result generalisable, due to the uniqueness of all organisations and all instances of decision situations. However, as presented in the previous chapter, we tried to get an indication of the transferability of the case study results with the help of four experienced IT consultants, who were interviewed in groups of two (see chapter 5 for methodological details). Thus, we can conclude that the results of the case study are not exclusive for just this particular organisation.
The discussion with the IT consultants primarily focused on the problems and difficulties identified in the case study and not on the factors per se. Concerning the problem of the low status of requirements work, the IT consultants thought this to be rather rare. Instead, according to their experience requirements engineers often have high status and are often skilled and knowledgeable persons (see Quotation 119 - Quotation 121). The problem of prestige between subsystems was confirmed by the experience of the IT consultants (see Quotation 122).

Quotation 119: “What doesn’t agree with my experience is this with persons fresh out of university, callows, which have this job. Instead, I often see old foxes in that job…”

Quotation 120: “No, fresh out of university, I haven’t come across that.”

Quotation 121: “However, I don’t feel this to be a work of low status, so to speak. Instead, I would say that that they have high status, those who work with… It is they who make the decisions about how it should be done. And they are most often very knowledgeable. You rarely come across someone who is inexperienced and doesn’t know anything, so to speak.”

Quotation 122: “Between different subsystems, I believe it [i.e. prestige] is, more or less. Some more, others less.”

Concerning the problems and difficulties related to the factor communication and coordination, the IT consultants said that these agreed with their experience of other development organisations (see Quotation 123 - Quotation 125).

Quotation 123: “… this, that the requirements are written in different ways by different persons, it has happened to me, which makes them difficult to interpret or that they haven’t been possible to verify…”

Quotation 124: “A requirements engineer is often viewed as an administrative, non-technical person or role and they do not take part when technical decisions are discussed and things like that. So, I think it is correct that they aren’t involved [i.e. in discussions].”

Quotation 125: “I would say that if they, who write the requirements, could communicate better then they [i.e. the requirements decisions] would land in the implementation phase and in the test phase at the same time. […] I think that the requirements engineers have a possibility to disseminate information to so many people as possible when the requirements have been updated, so that they reach everybody at the same time.”

The IT consultants felt the problems and difficulties relating to the factor resources were difficult to relate to. However, they said they believed it to be correct although, to a large extent, this varies between different companies. Concerning the problems and difficulties of the factor pressure, they all claimed it to be right. Finally, with regard to the problems and difficulties of the factors cognitive load as well as knowledge, the IT consultants said that this partly agrees with their experiences.
Furthermore, this varies between different organisations (see Quotation 126 and Quotation 127).

Quotation 126: “They can rather easily manage this part, the system people I mean. However, in other projects that I have taken part in, this is a large problem, where you don’t know how the system actually works.”

Quotation 127: “Well, I think this also varies a lot. Because one has met persons that clearly have no idea of what requirements they write. At the same time, you have seen requirements engineers that are very skilled and have technical knowledge of how everything works. I think this is something that varies.”

Thus, we can assume that the factors are not unique for the company in the case study, and that they can be found in many development organisations. We can conclude that when one or more of these problems and difficulties related to the factors appear in a development organisation, then potential quality problems in the RE decision processes exist.

Altogether, there are several difficulties to manage in order to be able to improve RE decision-making. A way to do this is to provide better requirements engineering decision support that is adapted to the needs of the requirements engineers. RE is special and so is RE decision-making (see Quotation 128).

Quotation 128: “I would like to say that I experience requirements engineering as very much being a mind set, that you have to think in a certain way that you don’t do in other closely related processes. You have to have another focus.”
9 Desirable characteristics of RE decision support systems

In this chapter, we present desirable high-level characteristics of an RE decision support system (REDSS). Our working definition of an REDSS is as follows: An REDSS is a computer-based information system that supports either a single RE decision-maker or a group of RE decision-makers when dealing with unstructured or semi-structured RE problems in order to make more effective decisions. The REDSS supports one or more RE decision-making activities carried out in an RE decision process.

For clarification, in this thesis we use the term REDSS to refer to a visionary, non-existing tool and the term RE tool represents existing tools.

The characteristics can be viewed as a wish list and the more characteristics that are fulfilled the more effective and efficient the REDSS will be. For each characteristic a guiding principle or principles that can be used as guidelines for how to fulfil it are suggested. In addition, some available techniques for each guiding principle are put forward (see Figure 30).

9.1 Characteristics, guiding principles, and techniques

Together, the desirable characteristics and their guiding principles constitute a visionary REDSS, the potential realisation of which is demonstrated by the available techniques.

The desirable characteristics are based on the needs of the RE decision-makers in order to positively influence the factors that affect the outcome of RE decision-making. In addition, the desirable characteristics are based on the nature of the activities in the RE decision processes - not on RE activities and tasks per se. This means we do not directly address RE decision matters such as choosing the most appropriate elicitation technique or prioritising. Instead, we focus on the generic
human activities that take place in the RE decision processes, for example, problem solving, communication, and idea generation. While the generic human activities are independent of the RE context, such as RE maturity level or type of application, RE activities and tasks as such are more context dependent. Although, they are important, the RE activities and tasks are beyond the scope of this thesis. The characteristics represent how an RE decision-maker will experience the REDSS. The suggested REDSS characteristics and their related guiding principles are summarised in Table 2, in which we refer to the sections where they are elaborated. Furthermore, the table also shows the derivation from the empirical findings concerning the decision situation of RE decision-makers.

As the desirable characteristics, the guiding principles are empirically grounded. However, in contrast to the characteristics, the principles are also theoretically grounded. This means that we base our suggestions on both the case study findings and theoretical knowledge. The purpose of the guiding principles is to direct further efforts concerning how to find a solution. Hence, we do not claim that the principles are the solution, but a possible way to find a suitable solution.

The term technique has a broad meaning here. Actually, while we sometimes present techniques, they can also be design strategies, design principles, approaches etc. What all these “techniques” have in common is that they represent a way of elaborating the principles and that they show that it should be possible, in some sense, to realise the visionary view of REDSS. The techniques presented together with the guiding principles should be regarded as a range of potential means.

It is beyond the scope of this thesis to contribute with specific user interface design suggestions or technical solutions for how to implement the techniques.

Table 2, Derivation of the suggested characteristics and guiding principles

<table>
<thead>
<tr>
<th>Problems in RE decision-making</th>
<th>Nature of RE tasks</th>
<th>Desirable characteristics of REDSS and their related guiding principles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expressed problems concerning the cognitive load is:</td>
<td></td>
<td>Reduce the cognitive load (see section 9.2.1)</td>
</tr>
<tr>
<td>• Lack of general overview</td>
<td></td>
<td>• Present both overview and details</td>
</tr>
<tr>
<td>• High memory load</td>
<td></td>
<td>• Provide memory aid</td>
</tr>
<tr>
<td>One of the problems related to the factor resource is:</td>
<td></td>
<td>Ensure high usability (see section 9.2.2)</td>
</tr>
<tr>
<td>• Usability problems in requirements management tools</td>
<td></td>
<td>• Follow usability design principles</td>
</tr>
<tr>
<td>RE decision-makers handle information in different ways, e.g., searching, creating, and analysing. Different types of information (see section 9.3.1)</td>
<td></td>
<td>• Apply information</td>
</tr>
</tbody>
</table>
| Information sources are used, e.g., requirements and requirements-related information, The Internet, as well as records and reports. The information is often stored in databases or as documents. | visualisation on different levels of use  
- Combine data-driven and document-driven DSS techniques |
| --- | --- |
| Different types of decisions are made; some are main decisions and others are sub decisions. Some decision matters are system-related and others are work-related. | Support different types of decision matters (see section 9.3.2)  
- Integrate requirements decision support techniques  
- Integrate RE process decision support techniques  
- Integrate requirements-based decision support techniques |
| The process of establishing requirements involves many creative challenges. | Support creativity and idea generation (see section 9.3.3)  
- Integrate creativity enhancing techniques |
| There are knowledge-related problems concerning the domain, the product, and requirements engineering. | Support knowledge sharing and transfer (see section 9.3.4)  
- Apply knowledge management approaches |
| The process of managing requirements changes is initiated by, e.g., requirements errors and change proposals, which calls for idea evaluation and problem solving. | Support idea evaluation and problem solving (see section 9.3.5)  
- Integrate evaluation approaches  
- Integrate cognitive tools to augment human problem solving |
| There are communication problems:  
- Little involvement in discussions  
- Little communication of decisions | Decision communication activities are carried out during the RE decision processes; communications with stakeholders, dissemination of decisions, and negotiations. | Support decision communication (see section 9.4.1)  
- Provide additional communication paths  
- Provide negotiation facilities |
| There are coordination problems:  
- Lack of coordination of way of working  
- Time-consuming coordination | Some RE decisions concern coordination. | Support coordination (see section 9.4.2)  
- Integrate coordination technologies |
The characteristics are structured using Figure 31. Firstly, the characteristics of the support relating to the RE decision-maker as a user are described. The characteristics originating from the nature of the decision-making tasks within the RE decision processes are then presented. Finally, the characteristics that support the RE decision-maker in the social context are introduced.

Within each characteristic, we begin by explaining and giving motivations for it. We then introduce the guiding principle or principles for the characteristic. As with the characteristic, we start by describing each principle and give reasons for it. The relation between the principle and its characteristic is included. Next, some available techniques for each principle are introduced. These techniques are described and motivations are given, e.g., their relation to their guiding principle and characteristic. The potential consequences for RE decision-making are frequently discussed and illustrated with an example. On occasion, we include empirical quotations to illustrate some aspects. The examples and quotations are highlighted with the help of indentation and font size.

9.2 The RE decision-maker as user

The RE decision-maker is a user of the REDSS, and a user perspective is vitally important for REDSS to be highly beneficial. If users experience a supportive, smooth and pleasant tool, the chances of frequent and appropriate use of the REDSS will increase. However, if basic user needs are ignored, then the REDSS will be less valuable and too cumbersome to use.

Two identified characteristics of an REDSS from the perspective of the RE decision-maker are:

- Reduce the cognitive load
- High usability

9.2.1 Reduce the cognitive load

The REDSS should reduce the cognitive load of the RE decision-maker. Cognitive load is a “construct representing the load that performing a particular task imposes on the cognitive system” (Sweller et al., 1998, p 266). In other words, cognitive load
means a mental exertion, the purpose of which is to interpret and process information in order to decide an action within a given space of time (Gulliksen & Göransson, 2002). Our mental capacity to consciously process information, which is called controlled processing, is limited. Thus, it is important that the individual uses this limited capacity for the most important and mentally demanding tasks (Schneider, 1993).

One of the factors that has an effect on the RE decision-maker is the cognitive load, which we identified in the case study (see section 8.5). The empirical study showed that the RE decision-makers sometimes experience difficulties and problems that originate from a high cognitive load (see Quotation 129 and Quotation 130).

Quotation 129: “We have about 500 pages on [subsystem X], I think it is 450 on [subsystem Y] and about the same on [subsystem Z]. That is too much actually, because nobody is able to go through it. […] If you have worked with it a while, then you get this overview. However, it takes a long time. I have worked as a requirements engineer for [subsystem Y] for a year now. I don’t have an overview of it.” (Requirement engineer)

Quotation 130: “However, unfortunately, this tool [i.e. a requirements management tool] makes them [i.e. the requirements] end up higgledy-piggledy. So it becomes very hard to read it through and get a grip of it.” (Requirement engineer)

This means that the conscious mental information processing of the RE decision-makers is affected. As a consequence, the quality of information interpretation and decision-making can decrease.

For example, a lowered cognitive load makes it possible for the RE decision-maker to concentrate on the decision activity at hand, e.g., decide upon if the requirements become baseline, instead of using his or her limited mental capacity to consciously process information on unnecessary tasks, e.g., trying to recall requirements dependencies from the (human) memory.

Hence, unnecessary cognitive load should be avoided.

Two guiding principles are suggested in relation to this desirable characteristic. To reduce the cognitive load, the REDSS should

- Present both overview and details
- Provide memory aid

**Present both overview and details**

One way to manage the cognitive load is to present both overview and detail (Schneider, 1993). When a decision-maker has a large amount of information to take into account, the information should be presented in a way that facilitates the cognitive information processing. If the information details are not related to some kind of whole or not presented in a context, then it will be difficult and time-consuming for the decision-maker to interpret the information and judge the
importance of it in relation to the decision at hand. Hence, the cognitive load will be too high.

One of the expressed problems concerning the cognitive load in the case study was that the RE decision-makers experience a lack of overview (see section 8.5).

For example, the RE decision-maker should always be able to obtain a global view of the system and the dependencies that exist within it. It should be possible for the RE decision-maker to understand the relationships between, on the one hand, a requirement and its details, and on the other hand, the system, other requirements, the system engineering process, or other relevant contextual information. A comprehensive understanding of a requirement in relation to a whole makes it possible for the RE decision-maker to judge the consequences of a certain requirement decision.

Hence, to put it in the words of Card (2003) and Spence (2001), in order to understand the information details, the user needs to see the details in their context to be able to do a plausible interpretation.

Some available techniques

The field information visualisation provides design strategies and techniques to manage the problem of presenting both overview and details so that the cognitive load of the RE decision-maker is reduced. Information should be visualised in a way so that the receiver, i.e., the RE decision-maker, can grasp it easily. Information visualisation is defined as “the use of computer-supported, interactive, visual representations of abstract data to amplify cognition” (Card, et al., 1999, p 7). Hence, the primary purpose is to support and augment human cognition. Card, et al. (1999) list different ways that information visualisation can support cognition. For example, the human memory capabilities and the information processing capacity can increase through externalisation. The RE decision-makers frequently use abstract data, e.g., requirements, in RE decision-making. If this abstract data is represented interactively and visually in the REDSS, then the cognitive capabilities, including decision-making capability, of the RE decision-makers should be augmented. This means that the RE decision-maker should be able to choose and manipulate the visual views in the REDSS.

For example, when preparing for a requirements negotiation session, the RE decision-maker may want to get the overview of previous arguments from stakeholders together with the details of the current requirements. The RE decision-maker may, for instance, shift between different stakeholder views.

Tufte (1990) presents a number of design strategies that facilitate the communication via a two-dimensional space. The main challenge is to communicate complex data on a limited area, without losing information and at the same time making it easy to use and understand. Tufte stresses that a good design is transparent and gives focus to the information that is to be mediated. If the design elements dominate, the attention is drawn from the information and the communicative power decreases. Design elements are “things” added in the design space that are not the information itself,
e.g., a line separating text sections, a warning symbol, fonts, or a coloured background. For instance, an appropriate font increases the readability of a text message, while a flourish font decreases the readability of the text. The design strategies that Tufte advocates are:

- Micro/macron readings
- Layering and separation
- Small multiples
- Colour and information

The design strategy behind micro/macron readings is to add details. Details that together create a coherent structure. Wisely organising the details forms a whole and a context that makes it easier for the receiver to perceive and comprehend the complex information. The overview and the detail are thereby delivered in the same picture. By using the empty space in an efficient way the risk of information overload is reduced despite the large amount of information. This way, the user can make global comparisons without changing context (Tufte, 1990). A simple example of the design principle can be seen in Figure 32.

![Figure 32, An example of the design strategy micro/macron readings](image)

In this example, the requirements, i.e. the details or micro readings, form an overview, i.e., macro reading, by grouping related requirements using the empty space, distance, and design elements (lines and a square).

For example, if an RE decision-maker has a particular interest in the fourth version of usability requirement 6 for subsystem X, then related information concerning other requirements is visually represented. From this overview, the RE decision-maker can explore the information needed for the decision-making task at hand. The overview makes the RE decision-maker aware of the problem space that needs further analysis.

By visual separation of different data types it is possible to create information layers (Tufte, 1990). All elements in a design interact with each other and this visual effect can be described as $1+1=3 \text{ or more}$ (Albers, 1969 in Tufte, 1990). Tufte claims that one way, among others, is to create a layer by colour coding a certain data type. For example, if we have a detailed model of a machine, where all the parts of the machine are coded in black and their attributes in red, then the machine parts will constitute one whole – a layer – while the attributes will be related to each other –
another layer – and still, the model is not messy and overloaded (Tufte, 1990). Tufte’s example can be transferred to RE.

For example, the RE decision-maker may need information concerning the graphical user interface standards in order to make a particular decision. A prototype of a representative user interface can then be visualised, which forms a layer. Another information layer can be placed on the prototype in the form of textual descriptions of the standards. If the text is, e.g., in light grey, then the two related information layers can be viewed together, hopefully without disturbing each other. Overview and details are then presented together, which in turn can reduce the cognitive load.

The purpose of the design strategy small multiples is to support comparisons between different data sets. It is concerned with bring about differences and to compare changes within the data. In order to make an effective comparison, the information has to be uninterruptedly present within the scope of the eye span. Thus, small multiples denote that the range of alternatives should be revealed at the same time (Tufte, 1990).

For example, if a group of RE decision-makers is going to decide upon which system requirements belong to which subsystem, then it may be useful to be able to see all system requirements within the scope of the eye span with the help of the REDSS and a projector. The system requirement may then – in the same view – be dragged and dropped in subsystem “spaces” of the user interface. Different alternatives can then be look at and discussed.

Colour can be used for coding abstract information. While colour coding can be effective, Tufte (1990) warns that inappropriate use can be harmful. Thus, colour should be used cautiously. There are four elementary uses of colour in information design: a) labelling, b) measuring, c) representing or imitating reality, and d) enlivening and decorating. Colour as labelling means that colours can symbolise different types of information so that the user can distinguish them. Darkening can visualise measuring. Furthermore, colour can be used to more explicitly relate to the reality that is represented. Finally, colour can liven up the data sets and be beautiful (Tufte, 1990).

For example, colours can be used for coding abstract RE information as well, e.g., status, priority, information owner, or classification. In an REDSS, the RE decision-maker should be able to interact with the data sets and adjust the colour coding in order, for example, to highlight information that is particularly relevant for the decision at hand. For example, colour may help to organise information so that patterns can be revealed. This way, the RE decision-maker can conduct diagnosis work and, for instance, create a general view of the needs and problems in the systems engineering process to come.

There are also different techniques to use so that the user can see both overview and details at the same time and thereby reduce the cognitive load of the RE decision-maker. These techniques address the problem of focus + context, by presenting two separate views together. An example concerning large documents is to present
miniatures of pages or a document map, i.e., the context, together with the current page, i.e., the focus (Spence, 2007). This can be used in the REDSS for requirements documents, for example, when breaking down requirements in a requirements specification to a function specification. The RE decision-maker will then, e.g., see an overview of related groups of requirements in a document map while working with particular functions.

Two other examples of concrete focus + context techniques are the DragMag Image Magnifier (Ware & Lewis, 1995) and Magic Lens (Spence, 2007). DragMag and a Magic Lens provide flexible positioning on a small region of interest to be magnified, often with added information details, although in somewhat different ways (Spence, 2007). For the RE decision-maker, these techniques can be useful, for instance, to show both overview and details of complex graphical models.

For example, if a goal model is visualised in the REDSS, then the user can position the magnifier onto a certain goal in the model and receive detailed information of that particular goal. This way, the RE decision-maker can gain focused goal information without losing its context. An additional advantage is that the model is not cluttered with details, since they are hidden in the context area.

In sum, if the design strategies and focus + context techniques presented above are taken into account and implemented in an REDSS, the difficulties RE decision-makers have when obtaining a general view can be reduced. A general overview facilitates the understanding of, e.g., a subsystem, its internal characteristics, as well as relationships and dependencies within and outside the subsystem. The details of requirements information, important for a current RE decision, are contextualised with the help of these design strategies and focus + context techniques. This makes it easier for the RE decision-maker to properly interpret the information, and hence, can directly have a positive effect on RE decision-making.

The presentation of overview and details together can reduce the cognitive load of the RE decision-maker. Another guiding principle can also be derived from the case study of decision situations for RE decision-makers. This principle is to provide memory aid.

**Provide memory aid**

Another guiding principle that aims to reduce the cognitive load is to provide memory aid, which makes it easier for the RE decision-makers to externalise aspects that otherwise have to be remembered and recalled when needed. An example of external memory is storing rationale for previous requirements decisions. Ashcroft (2006) explains that the human memory plays an important role in human information processing. Recall of information from the human memory requires mental exertion and therefore affects the cognitive load. Thus, by providing memory aid, unnecessary cognitive load can be reduced.

One of the expressed problems concerning the cognitive load in the case study was that the RE decision-makers experience a high memory load (see section 8.5).
Important aspects, that are relevant for RE decision-making, have to be kept in the mind of the RE decision-makers. By providing memory aid, this problem can, at least to some extent, be avoided.

Some available techniques

The human memory can be supported, for instance, by a decision support system function called Case Memory (Chen & Lee, 2003). Case Memory lets the decision-maker manage “soft” information of a business case such as personal experiences, rumours, and opinions of others. The cases can be private for the user or public for all in the decision-making team. In order to facilitate capturing of soft information, voice recorders can be a complement (Chen & Lee, 2003). The Case Memory provides a particular kind of memory aid, i.e., information concerning cases, which in turn can reduce the cognitive load of the RE decision-maker.

For example, this can be in a decision process of release planning. It can then, for instance, be useful for an RE decision-maker to easily record his/her impressions and gut feelings after having discussions with external stakeholders. It may be useful later on to be able to recall this information in a correct way, since relying on the human memory can be risky.

There are additional cognitive advantages by using Case Memory according to Chen and Lee. It also reduces availability bias, which occurs due to the human tendency of using the availability heuristic, which means that humans judge the frequency of an occurrence based on how easily it comes to mind. However, how easily something is available in the human memory is not a good frequency estimate (Parkin, 2000; Reisberg, 2006). Case Memory helps the decision-maker recall contextual information and conditions of past events and decisions. The use of analogical cases enhances problem-solving and creative thinking (Chen & Lee, 2003; Parkin, 2000; Reisberg, 2006).

If such a function or similar functions are implemented in an REDSS the memory load and hence the cognitive load of the RE decision-maker can be reduced. As a result, the potentials of high quality RE decision-making are improved. In addition, high usability of the REDSS can also reduce the cognitive load.

9.2.2 Ensure high usability

The REDSS has to have high usability. Usability is the quality of the interaction between the user and the system (Benyon et al., 2005). Usability consists of several goals (Preece et al., 2002), which should be fulfilled by the REDSS: effectiveness, efficiency, safety, utility, memorability, and positive experience. It is important that the effectiveness is high, so that the result of using the REDSS is valuable. The REDSS should be efficient, since it would otherwise be too cumbersome to use, thus lowering the productivity of the work. In order to ensure that the REDSS is used in the right way and to prevent errors the safety should be high. It should also provide the functionality needed from the RE decision-makers’ perspective, i.e., it should have high utility. To lower the learning threshold, the learnability of the REDSS
should be high. Memorability, i.e., remembering with infrequent use, is probably also of importance from the RE decision-makers’ perspective, since there can be different usages of the REDSS. Some users will hopefully frequently use the REDSS. However, there can also be those who use it more seldom. The use of the REDSS should also be a positive experience. The subjective feelings towards the tool can, for example, be trustworthiness, satisfaction, motivation, and helpfulness (Preece et al., 2002).

One of the problems that the RE decision-makers in the case study have is that the requirements management tool suffers from usability problems, particularly low effectiveness, low efficiency, and low learnability (see Quotation 131 - Quotation 135).

Quotation 131: “It is perhaps that [the requirements management tool] it is relatively cumbersome. This is what it actually is all about. That it offers resistance to sit and work with it.” (Requirements engineer)

Quotation 132: “… it is the problem solving that is fun. To manage everything then in a requirements management tool… That is much harder.” (Requirements engineer)

Quotation 133: “Well, best possible observability. Then, there is the question of how easy the tool is to use. And there, you always want it to be more easy to use. More intuitive.” (System engineer manager)

Quotation 134: “That, I think is a strong wish. More intuitive handling.” (System engineer manager)

Quotation 135: “I think requirements engineering in database form or similar is absolutely essential and the next step is that it should be usable, easy to handle and thereby available for everybody affected by it. That, clearly, I think is most important.” (System engineer manager)

The usability problems of the tool give negative consequences, for instance, low penetration of the tool in the organisation (see section 8.3). This can be compared with “the two generally agreed necessary conditions for DSS success and […] the factors that lead to their achievement: thus success is being equated with repeated use and user-satisfaction” (Finlay & Forghani, 1998, p 54). Since REDSS is a DSS these conditions are equally important for an REDSS as they are for other decision support systems. An important factor that positively has an impact on frequency of use and user satisfaction is to make sure that a system has high usability (Benyon et al., 2005; Gulliksen & Göransson, 2002).

In order for RE decision-makers to experience high usability in the REDSS, the following guiding principle is crucial:

- Follow usability design principles
Follow usability design principles

There are usability design principles that should be followed in order to achieve REDSS with high usability. Design principles are “generalizable abstractions intended to orient designers towards thinking about different aspects of their designs” (Preece et al., 2002, p 20). Preece et al. further explain that a usability principle is more prescriptive than a design principle and can be used, not just for informing a design, but also for evaluating existing systems and prototypes. Thus, if usability design principles are used, then they can both guide the design efforts as well as have more prescriptive power with regard to ensuring high usability of the REDSS.

Some available techniques

Principles that are mainly concerned with access, learnability, and memorability are (Benyon et al., 2005):

- Visibility
- Consistency
- Familiarity
- Affordance

Visibility is about letting the available functions of the system (i.e. the REDSS) and the system’s process status be visible for the user (Benyon et al., 2005). If available REDSS functions, for instance, are grouped and presented in a task-based menu, then which tasks that are supported and which services the system provides for each task would be visible for the RE decision-maker.

For example, if a menu label is idea generation and another is impact analysis, then the RE decision-maker will know that the tasks idea generation and impact analysis can be supported and easily found in the menus of the available techniques.

Consistency states that design features should be used in a consistent way, both within the REDSS and with similar systems and relevant standards (Benyon et al., 2005). For example, if the terms idea generation and impact analysis are used in some part of the user interface, then exactly these terms should be used in all interactions between the RE decision-maker and the REDSS. Otherwise, it would be confusing for the user.

For example, if certain symbols are used to indicate the status of requirements, e.g., ‘requirements error’ or ‘ready for prioritisation’, then a particular status should always be presented with its symbol. Otherwise, there may be a risk that the RE decision-maker does not notice that a particular requirement needs attention.

Familiarity suggests that language and symbols familiar to the intended users, i.e. RE decision-makers, should be used. An appropriate metaphor can also enhance the knowledge transfer from a familiar domain to the tool (Benyon et al., 2005). This means that the REDSS should use terminology and other representation forms that is
commonly used by RE practitioners. Then, the RE decision-makers do not need to learn, e.g., new terms or notations. Familiar language and symbols are one way of lowering the learning threshold and reduce the risk of using the REDSS in an inappropriate way.

Affordance concerns the properties of a design feature which make it obvious how the item is to be used. The functions of the REDSS should be designed so that the user can perceive how to use them (Benyon et al., 2005; Norman, 1988). The RE decision-maker should not need to learn how to interact with the REDSS. Instead, it should be obvious for him or her in the current interaction situation how to manoeuvre the system. A trivial example is that if a user should want “click” on something, the icon or text should look like a button and not like something with just aesthetical or informative value. The perceived affordance of a button is to be clicked upon. With a natural perceived affordance of the functions in the REDSS, the RE decision-maker will most likely not spend so much time figuring out how to interact with the system.

These usability design principles increase the probability that the decision-supporting power of the REDSS is optimally utilised, since they all make it easier for the RE decision-maker to understand what services the system provides and how to make use of them. These usability design principles are important since they emphasise learnability and memorability. The case study showed that RE decision-makers can be novices regarding both RE, the domain, and the system to be (see section 8.6), which calls for learnability. Some RE decision-makers use RE tools infrequently, which causes a need for memorability.

However, for the REDSS to have high usability, not only learnability, memorability, and access should be addressed. Benyon et al. (2005) also list design principles that are primarily focused on the efficiency of the system, i.e., make it more easy to use. These are:

- Navigation
- Control
- Feedback

Navigation consists of maps, directional signs, and information signs. These enable the user to move around in the system (Benyon et al., 2005). The REDSS can provide navigation in different interaction spaces, e.g., navigation in the set of requirements and related documents, navigation within and between subsystems, or other interaction spaces that can be relevant in the REDSS.

For example, in the diagnosis routine of a decision process, the RE decision-makers have to grasp partially ordered information. They may need to initiate themselves to different information in various requirements documents in order to analyse what matters for the subsystem. Then, it should be easy to navigate from, e.g., requirements at a product level to requirements at a goal level, from there to a vision document and then further on to a release plan.
Control allows the RE decision-makers to be in charge of the interaction with the system. Clear and logical mapping between controls and their effects in the world augment the possibility of user control (Benyon et al., 2005). The RE decision-maker should not feel at the mercy of the REDSS. There should be a transparency so that the user knows why something happens.

For example, the REDSS should not automatically draw dependencies (if that could be possible) between requirements without control of the RE decision-maker. He or she should, in such a case, not know if the requirements dependencies are correct or relevant for the decision matter at hand. The user should be in control and must know that the REDSS can be trusted. Otherwise, he or she probably will avoid using it.

Feedback of the effect of the actions taken with the REDSS should be provided. The feedback ought to be constant and consistent (Benyon et al., 2005; Norman, 1988). Immediate feedback to the user is important for the usability aspect, efficiency. Otherwise, the user will feel insecure about what is actually happening and may do extra checks of, e.g., if a certain process is actually running. However, in the REDSS feedback should not only be given to the user who has performed the action. Other RE decision-makers may also need feedback.

For example, if one user of the REDSS has made changes in a requirements document, then it could be important for other RE decision-makers to also get feedback about these changes.

Navigation, control, and feedback together make the REDSS more efficient to use. This is particularly important for frequent REDSS users. If the system is inefficient and difficult to use on a regular basis, the RE decision-maker would most likely avoid using the REDSS. Then, the system will have low penetration in the organisation and, hence, the positive affects of a decision support system will not be reached.

Furthermore, there are design principles that primarily enhance the usability aspect safety as well (Benyon et al., 2005). These are:

- Recovery
- Constraints

Recovery from slips and mistakes should be facilitated in a rapid and effective manner (Benyon et al., 2005). The REDSS should also make it possible for the RE decision-maker to notice, diagnose, and recover from errors that are made.

For example, if an RE decision-maker formulates a requirements decision in a way that differentiates from a prescribed way for formulation (which may, e.g., cause wrong interpretation and therefore erroneous decision implementation), then the REDSS can notify the user of this discrepancy. He or she can then diagnose if the current formulation of requirements decision can trigger problems. To facilitate recovery from the mistake, the REDSS can suggest a new formulation.
Constraints prevent users from doing inappropriate actions in the REDSS. There are, e.g., physical, semantic, and cultural constraints that can be used (Benyon et al., 2005; Norman, 1988). A physical constraint makes certain actions impossible, e.g., choosing an unfeasible menu alternative. Semantic constraints are based upon the users’ knowledge of the world in general or the current situation, e.g., the necessity of conducting some specific actions in a particular order. Cultural constraints are grounded in cultural conventions, e.g., the cultural meaning of colours or socially accepted behaviours (Norman, 1988).

The usability design principles, recovery and constraints, prevent the RE decision-maker from making, especially severe, slips and mistakes and facilitate management of the inappropriate actions that have been taken.

All design principles mentioned above address usability aspects that can be objectively measured, for example, time taken to perform a certain task or the number of slips made during a particular task. However, the users’ subjective experiences are also highly important for a system in order to have high usability. Examples of subjective experience are that the system should be satisfying, helpful, motivating, or aesthetically pleasing (Preece et al., 2002). To augment a positive experience of the REDSS, Benyon et al. (2005) propose three design principles, namely:

- Flexibility
- Style
- Conviviality

Flexibility is about allowing several ways of doing things and letting the users personalise the system (Benyon et al., 2005). Individuals in a population of RE decision-makers are different from each other, as are people in general. For that reason, it should be possible to use the REDSS in a variety of ways and it should be possible to get the information presented in an individualised way. The REDSS should meet the needs stemming from cognitive style, decision style, and level of expertise.

All persons have their own cognitive style that affects their way of processing information. There are several dimensions and one of them is field dependency versus field independency. A person with a field dependent style tends to tackle a task in a holistic manner, avoid details and is more interested in the global picture. In contrast, the field independent individual is more serialistic when approaching a task, concentrates on details, and has a tendency to separate figures from their context (Witkin et al., 1977; Chen et al., 2005). Chen et al. (2005) argue that a flexible user interface is important in order to meet the different preferences stemming from cognitive styles. Their work on flexible interface design for web directories show, for instance, that it is supportive for a field independent user to get the main categories and subcategories sorted and listed on the same page. The field dependent user, on the other hand, finds it more helpful to have the main categories and subcategories presented on different pages. It can therefore be argued that an REDSS should also
provide support in a flexible manner according to different cognitive styles so that each RE decision-maker is supported in the most effective way. In the same way individuals have their own decision styles, which are related, for example, to preferred communication channels and problem solving strategies (Rowe & Boulgarides, 1992). This also implies a need for high flexibility in the REDSS.

The case study showed that RE decision-makers can be novices regarding both RE, the domain, and the product. Consequently, the REDSS must meet the needs of both inexperienced as well as experienced RE decision-makers.

For example, the way of interacting with the REDSS can differ. A novice may need support with what can be performed and how it could be carried out. More explanation and more instruction may be needed. An expert may instead want to interact through short cuts.

*Style* states that the design of the system should be elegant and appealing (Benyon et al., 2005). If the REDSS is not elegant and appealing, the RE decision-maker may get an impression of unprofessionalism and therefore perhaps not trust the system. If they do not trust it, they would probably avoid using the REDSS.

*Conviviality* means that the interaction between the system and the user should be pleasant, polite, and friendly (Benyon et al., 2005). This makes the REDSS smoother to use and the interaction more transparent. A transparent interaction lets the RE decision-maker concentrate on the RE decision-making tasks instead of on manoeuvring the system. The opposite, an awkward and irritating interaction draws the attention from the important tasks and is instead directed to managing the interaction with the REDSS.

If the REDSS has high usability, then there are several positive consequences. First of all, an REDSS that is not used definitely can not enhance the potentials of high quality RE decision-making. An REDSS with high usability increases the possibility of it actually being used. Another positive consequence is that high usability augments the chance of the system’s decision-supporting capabilities being correctly and efficiently utilised. Mistakes can then, to some extent, be avoided. A third consequence, which also is mentioned in the previous section, is that high usability reduces the cognitive load, since the decision-maker can use the conscious information process capabilities to carry out the decision-making activities and not to manoeuvre the REDSS.

In summary, from the user perspective of an RE decision-maker, we should reduce the cognitive load and make sure that the usability of the REDSS is high. However, these characteristics are not enough in order to have an REDSS with high decision-supporting capabilities. The decision-making tasks of the RE decision-makers also need also to be considered. The nature of these tasks calls for other desirable characteristics of the REDSS.
9.3 The nature of RE decision-making tasks

The user, i.e. the RE decision-maker, conducts several different tasks. In order for the REDSS to be highly helpful and valuable, it is important that the system provides support for the actual decision-making tasks, otherwise the REDSS will not be suitable for its purpose.

We identify five characteristics of the REDSS that originate from the nature of the decision-making tasks in the RE (see chapter 7).

- Support availability of different types of information
- Support different types of decision matters
- Support creativity and idea generation
- Support knowledge transfer and sharing
- Support idea evaluation and problem solving

9.3.1 Support availability of different types of information

A REDSS should support availability of different types of information. Availability not only means access to relevant information, but also mental availability. The information has to be smoothly and correctly interpreted and understood by the RE decision-makers. Thus, the needed information should be accessible with the help of REDSS and the information should be properly presented.

The case study showed that RE decision-makers use several sources of information in their decision-making activities. They use information that is or can be stored in databases, e.g., requirements, as well as information stored as ordinary documents, e.g., reports from investigations and records from meetings (see section 7.3).

In order to provide availability of relevant information to the RE decision-makers, we suggest the following two guiding principles:

- Apply information visualisation at different levels of use
- Combine data-driven and document-driven DSS techniques

Apply information visualisation on different levels of use

Information visualisation should be applied at different levels of use, because the RE decision-makers handle information in different ways, e.g., investigating documents to get an understanding of the problem, creating use cases and writing requirements, and analysing risks (see section 7.1). To support mental availability, the different levels of actual use should be taken into account. Otherwise, there is a risk that the information is provided in an unsuitable way making it unnecessarily difficult for the RE decision-maker to make effective and efficient use of it.

Not only should the different levels of use be taken into account to support the mental availability, but also apply information visualisation. The purpose of information visualisation is to amplify cognition, and by applying such techniques at the different levels of use knowledge crystallisation can be augmented (Card, 2003).
Card explains that knowledge crystallisation is a task which has a goal of some kind that demands meaningful information and embraces the creation of a knowledge product, decision, or action. The knowledge crystallisation process is iterative and consists of four parts (Card, 2003):

1. Acquire information – finding, identifying, and capturing relevant information
2. Make sense of it – externalising, organising, and arranging information so that it makes sense and becomes useful
3. Create something new – to reach the intended goal something has to be created, e.g., a report, through knowledge work, e.g., problem solving
4. Act on it – some activities can be necessary to reach the goal, e.g., distributing the report

RE decision-making can be said to consist of knowledge crystallisation tasks. Hence, information visualisation techniques can help the RE decision-maker find and make sense of the information needed for decision-making.

For example, when an RE decision-maker in the diagnosis routine investigates a requirements error report, the REDSS can provide support on different levels of use. First, the RE decision-maker can use the REDSS to find the related requirements documents and identify their relevant parts. This information can then be scrutinised in the REDSS by manipulating and visualising the relations between the information chunks, making it easier for the RE decision-maker to understand. After that, the RE decision-maker needs to formulate one or more alternatives for how to deal with the requirements errors, which then are decided upon. Finally the decision concerning the requirements error is distributed to the concerned stakeholders via the REDSS.

Some available techniques

This can be applied at different architectural levels in the REDSS, which coincide with the four levels of use for information visualisation (Card et al., 1999; Card, 2003). The levels are infosphere, information workspace, visual knowledge tool, and visually enhanced objects (Figure 33).

![Figure 33, Levels of use for information visualisation (adapted from Card, 2003)]
The *infosphere* provides access to information outside the REDSS, i.e., outside the immediate work environment. It can be information on the Internet or the company’s intranet (Card et al., 1999; Card, 2003). The case study showed that the RE decision-makers searched, e.g., on the Internet for concepts they found in the documentation and had difficulties understanding. There are also records and reports that may not be stored in the REDSS, but which are needed for decision-making activities. Thus, an infosphere can be useful in an REDSS.

An *information workspace* can be thought of as a desk that integrates multiple visualisations or other information sources or tools in order for the RE decision-maker to carry out certain tasks. It is a space where several visualisations are related to one or more tasks; a place to hold work in progress (Card et al., 1999; Card, 2003).

For example, a task that can be performed in an information workspace is impact analysis. Impact analysis is a complex activity that needs to analyse information from different sources. Thus, an information workspace can be appropriate for such a task.

The purpose of *visual knowledge tools*, is pattern detection and knowledge crystallisation. Such tools can organise information to expose patterns that are hidden in the data, allow information exploration by letting the user interact with and manipulate the visualisations of information, as well as allow visual calculations (Card et al., 1999; Card, 2003).

For example, it would probably be useful for the RE decision-maker to explore sets of requirements, where, for instance, requirements dependencies or requirements implementation costs are visualised.

*Visually enhanced objects* expose more information from some object in a genuine visual form (Card et al., 1999; Card, 2003).

For example, if the REDSS user wants more information about the hardware for the product to be, for example, then there can be a construction drawing of the object being visually enhanced.

With the help of these four levels of use, the availability of information needed for decision-making is enhanced. Mental availability is amplified especially, since proper information visualisation enables users “to get information fast, make sense out of it, and reach decisions in a relatively short time” (Gershon et al., 1998, p 9). Accordingly, the potential of high quality RE decision-making is improved. However, the mental availability is not enough, the information needed should also be possible to access via the REDSS.

**Combine data-driven and document-driven DSS techniques**

The REDSS should combine data-driven and document-driven DSS techniques, so that the RE decision-makers have access to the information needed in the RE decision processes. A basic assumption in the field of decision support systems is that “good information is likely to improve decision making” (Power, 2002, p 1). Of course, this
assumption is directly passed on to REDSS. By making relevant information available to the RE decision-makers, RE decision-making is expected to be more effective and efficient. The case study showed that the information needed for RE decision-making is primarily stored in databases or as documents.

For example, in the analysis mode of the evaluation-choice, RE decision-makers trace the requirements to higher level requirements. The relevant information to conduct this analysis can then be expressed requirements stored in a database as well as a vision document. The analysis is facilitated if both the higher level requirements and the vision documents are easily available via the REDSS.

Both types of information are important in the RE decision-making activities, and both should thus be supported in REDSS.

Some available techniques

To use the DSS categorisation of Power (2002), the REDSS should be a hybrid system that combines data-driven DSS with document-driven DSS. The REDSS must be tailored so that the system provides the information needs and the decision support.

Data-driven DSS emphasises access to large amounts of structured data and support data analysis. The decision-maker can use the REDSS to identify facts and draw conclusions about relationships. A data-driven DSS includes tools that facilitate “drill down” for more details and “drill up” to get a broader, summarised view. Such summaries support predefined decision support needs. It should also be possible to “slice and dice” the data dimensions (Power, 2002). Power describes the characteristics of the DSS data in the following way. The data can be integrated from several databases. In addition, the DSS data can have a time stamp and historical data should be possible to view. Compared to operating data in transaction systems, the DSS data is non-volatile. The data should have multiple dimensions. The last of Power’s DSS data characteristics is the significance of enabling the development and maintenance of metadata.

For example, in an REDSS the time stamp and the viewing of historical data can be requirements version management in order to enable some types of traceability.

The purpose of such a document-driven DSS is to gather, retrieve, classify and manage unstructured documents (Power, 2002). A search engine is a powerful decision-aiding tool in a document-driven DSS. Such a DSS supports decision-makers in analysing, displaying, and manipulating text (Power, 2002).

For example, when deciding what a system is going to look like, the RE decision-maker needs different kinds of information stored as documents, e.g., scenario descriptions and graphical user interface standards of the product family. The REDSS should make these easily available.

Baker et al. (1998) propose important concepts that should be integrated in a system for complex document search which supports decision-making. These are:
With regard to priming, the decision-maker is exposed to stimulus material that can assist the person recognise and recall information that might be relevant.

For example, if an RE decision-maker for an in-house system has been informed of large organisational changes, then he or she needs to find out how they can affect that particular system. If the RE decision-maker is exposed to an overview of the contents of relevant documents in the REDSS, information exploring is facilitated. In this way, the problem space to be dealt with can be identified more easily.

A logical view provides the decision-maker with access only to the pieces of text that are believed to be relevant in a specific decision situation, i.e., the text is customised to the current information needs. Hierarchies should be used to allow multiple levels of abstraction.

For example, if the RE decision-maker is going to have a discussion about requirements decisions with a certain group of stakeholders, then he or she may need to prepare information in advance so that the decision meeting will be effective and efficient. The REDSS can then provide advanced support in compiling the relevant information for exactly this situation and facilitate getting a logical view of it.

Hierarchies increase the cognitive compatibility and they support organisation, comprehension, communication, and learning of complexity. The simple graphical form of lists should also be integrated in the system, since they support the human memory by facilitating recall. Finally, to promote the self-reflexive thinking of the decision-maker, a notational facility should be offered. Such a facility can be compared to a word processor that makes it possible for the decision-maker to copy and paste texts into notes (Baker et al., 1998).

For example, the REDSS can make it possible to mark information chunks and easily drag and drop a copy of them in a note pad. In this way, the RE decision-maker can explore the requirements documents and other types of relevant documents, e.g., test reports, to find, for example, ready-made solutions to reuse. Looking for ready-made solutions is a decision-making task in the search routine of a decision process.

The REDSS as a hybrid system combining data-driven and document-driven DSS techniques makes it more likely that the needed information is available. If Baker’s (1998) concepts for complex documents search are added, then additional decision-supporting capability is provided in the document-driven part of the REDSS.
Thus, if the RE decision-maker has the relevant information concerning the decision at hand, the possibility of high quality RE decision-making increases. However, the information should not only be accessible via the REDSS, it should also be mentally available, i.e., the RE decision-maker should be able to find and make sense of it easily.

The kind of information that should be used in RE decision-making depends on the decision matter currently being handled. The REDSS should not just make the information available, it is also important to support different types of decision matters.

### 9.3.2 Support different types of decision matters

The REDSS should support the different types of decision matters dealt with by the RE decision-makers, e.g., which system requirements belong to which subsystem; how is the subsystem going to behave and what is it going to look like; or in which order will the requirements be implemented (see section 7.1). The REDSS should not only support the main decision matters but also the sub decisions, since the sub decisions have consequences on the main decisions as well as the development process. It should support both system-related and work-related decisions. Different kinds of decision matters require different decision support techniques.

Based on a literature analysis of requirements engineering decision support techniques, we have identified three different types of RE decision support techniques, which should be integrated in the REDSS.

The following guiding principles should thus be applied:

- Integrate requirements decision support techniques
- Integrate RE process decision support techniques
- Integrate requirements-based decision support techniques

### Integrate requirements decision support techniques

Decisions concerning the requirements as such are supported with requirements decision support techniques. Such decisions concern, for instance, requirements prioritisation and choosing requirements in requirements negotiation. The types of decision matters supported by such techniques are primarily system-related decisions, i.e., the decision has an effect on the system in some way.

Some available techniques

In the RE field we find several requirements decision support techniques, of which some are probably already integrated in RE tools available on the market. This is exemplified by techniques that support decision-making with regard to the priorities of requirements. Karlsson et al. (1998) conducted an evaluation of methods for prioritising software requirements, and they found that the analytic hierarchy process (Saaty, 1980), abbreviated as AHP, was the most promising method. Another example of a requirements decision support technique is the work of Pomerol (1998),
which aims to support the RE decision-maker in the requirements analysis phase. The support is provided through scenarios. A scenario is regarded as a branch in a decision tree and Pomerol (1998) presents a way of reducing the complexity and proposes ideas for scenario management. A third example is the decision-making methodology developed by Rosca et al. (1997), which supports analysing and extracting the requirements that fall into the category of business rules. A fourth example is the support for selecting requirements called Quantitative WinWin developed by Ruhe et al. (2002). Quantitative WinWin is intended to be used in requirements negotiation and is based on the preferences of stakeholders, the business values of requirements, and a given maximum development effort.

Above all the effectiveness of system-related decisions can improve by integrating requirements decision support techniques in an REDSS. An additional consequence can also be an increase in the efficiency of RE decision-making tasks concerning requirements. However, in order to afford decision support of the RE process, we need to integrate RE process decision support techniques.

**Integrate RE process decision support techniques**

RE process decision support techniques assist RE decision-makers in *determining suitable ways to carry out the RE process*. Such decisions concern, for instance, choosing an acquisition method or choosing other RE techniques so that they suit the project at hand or the particular organisation. This means that these techniques are more focused on work-related decisions, instead of system-related ones.

**Some available techniques**

An example of an RE process decision support technique is the ACRE framework (ACquisition of REquirements), developed by Maiden and Rugg (1996). ACRE includes guidelines that are to be used when choosing acquisition method. Another example is the work of Jiang and Eberlein (2003) and Jiang et al. (2004) which resulted in a methodology as well as a framework that support RE decision-makers tailor the RE process and decide on suitable RE techniques for a certain project. A third example is the MODDE methodology (Model of Decision support system Design and Evaluation), developed by Meikle (2002). MODDE uses concepts from the legal domain and provides support in the requirements specification process. The concepts guide the RE decision-makers’ considerations in the decision-making environment.

A related approach is method engineering in which techniques that assist RE decision-makers in determining suitable ways to carry out the RE process can be found. Method engineering is “the engineering discipline to design, construct and adapt methods, techniques and tools for the development of information systems” (Brinkkemper, 1996, 276). Method engineering concerns process tailoring which is important in RE, as in all other systems engineering parts (Ågerfalk & Ralyté, 2006). With situational method engineering, it is possible to construct project-specific methods ‘on the fly’ (Mirbel & Ralyté, 2006).
Integration of RE process decision support techniques focuses on work-related decisions and makes the RE process more efficient. Furthermore, using the “right” RE techniques increases the potentials of making more informed system-related decisions. There are also decisions that go beyond the RE part of systems engineering. Other types of activities, e.g., testing, use requirements as an input for their decisions. Hence, additional decision-supporting capability is made available if we also integrate requirements-based decision support techniques in the REDSS.

Integrate requirements-based decision support techniques

The primary purpose of requirements-based decision support techniques is to use the existing requirements as input, for instance, in software engineering decision-making that can be aided by software engineering decision support (SEDS). SEDS is decision support for the whole life cycle of software engineering and evolution. Decisions concern processes, products, tools, methods, and techniques (Ruhe, 2003a). The requirements-based decision support techniques go beyond the RE process and perhaps also beyond the work of RE decision-makers. Nevertheless, such techniques can be regarded as an extension of the REDSS.

Some available techniques

An example of a requirements-based decision support technique is presented by Maiden et al. (2002). This work integrates the multi-criteria decision-making technique AHP (Saaty, 1980) and a requirements modelling technique called i* (Yu & Mylopoulos, 1994). In this way, trade-offs can be dealt with and decisions concerning which system architecture and design is most appropriate compared to the systems requirements can be supported. Another example is supporting commercial-of-the-shelf (COTS) decision-making. Alves and Finkelstein (2003) support the choice of COTS features. The support involves comparing and reaching the best balance between the customer requirements and the COTS constraints. A third example focuses on supporting software release planning. An approach for such support is provided by Greer and Ruhe (2004). This approach, called EVOLVE, is evolutionary and iterative. Based on a set of requirements, including their effort estimations and their categorisation into priorities by stakeholders, candidate solutions are generated from which the decision-maker can choose. A forth example is the decision support method proposed by Svahnberg et al. (2003), which is a multi-criteria decision method that supports the choice of architecture based on quality requirements.

An integration of requirements-based decision support techniques can give bonus effects on decisions made in the systems engineering process that are not concerned with requirements as such or the RE process. Such techniques can make the systems engineering process more successful in terms of being cost efficient and resulting in a system that satisfies the customers and/or the users. Of course, both requirements decision support techniques and RE process decision support techniques have the same potentials, since RE is an intrinsic part of systems engineering.

The boundaries between the three categories of RE decision support techniques mentioned above are not firm, e.g., decisions concerning requirements may also be
based on requirement. However, the techniques found in the literature primarily address one of the problems that characterise the categories.

The more RE decision support techniques included in an REDSS, the higher the decision-supporting capability of the system, since more different kinds of RE decision matters are supported. Another challenge facing RE decision-makers is being creative and generating ideas, so that new solutions and decision alternatives can be formulated.

9.3.3 Support creativity and idea generation

The REDSS should support creativity and idea generation. Creativity can be defined as a personality trait or an achievement. While the personality trait is regarded as a dispositional variable of an individual that leads to the production of acts, items, and instances of novelty, the achievement is the creative product of a process (Eysenck, 1995; Forgionne & Newman, 2007). Creativity is not highly dependent on individual traits, but instead can be learned and improved (Forgionne & Newman, 2007; Turban et al., 2005).

The process of establishing requirements includes many creative challenges for the RE decision-maker (see section 7.1.3), e.g., generating innovative solutions to user needs that can be transformed to requirements. This may need support. Forgionne and Newman (2007) conclude, based on experiments, that creativity enhancing decision-making support systems improve both the process of decision-making as well as the outcome of decision-making compared to both traditional DSS and no decision aid. Thus, the RE decision-makers would most likely also benefit from creativity enhancing tools in the REDSS. Such decision support can be directed towards reducing the effects of human decision-making weaknesses or cognitive limitations in general, and stimulate the perception, imagination, and creative insights of the decision-maker (Holsapple & Whinston, 1996; Silver, 1991).

In order to support the creativity and idea generation of the RE decision-makers, we suggest the following guiding principle:

- Integrate creativity enhancing techniques

Integrate creativity enhancing techniques

To enhance creativity and idea generation, the rigid thinking that blocks the generation of ideas needs to be broken down (Marakas, 2003). There are multiple techniques available to tear down the blockages, enhance creativity, and increase the number of ideas generated. Such techniques should be integrated in the REDSS.

For example, to be competitive in the commercial market a company that develops software-intensive products must generate innovative features. The RE decision-makers can then use the creativity enhancing techniques of the REDSS to increase the number of generated ideas. This in turn, can increase the possibility of some idea or ideas being novel enough to lead to a new competitive product.
Some available techniques

Marakas (2003) presents four categories of creativity enhancing techniques that can be used for decision support mechanisms in an REDSS. The categories are:

- Serendipity
- Free association
- Structured relationships
- Group techniques

**Serendipity** concerns the appearance of random events that make a person think in new lines. These are moments that just happen and cannot be planned for. However, a person can expose him or herself to new information that might enhance creativity, such as browsing libraries or visiting conferences (Marakas, 2003). Beale (2007) suggests a way to support serendipity by using ambient intelligence to augment user exploration for data mining and web browsing. Ambient intelligence means that information from the ambient environment and the history of interaction is collected to generate a more effective interaction between the system and the user. The systems are based on artificial intelligence and visualisation techniques, which identify what is interesting for the user and then support him or her in making new discoveries (Beale, 2007).

For example, an RE decision-maker who has elicited user goals that demand innovations in the products in order to be fulfilled. The RE decision-maker can then use the serendipity support in the REDSS to browse the Internet and be exposed to related information that can break down rigid thinking and lead to novelties which in turn can bring about effective requirements decisions.

Such **free association** techniques focus on two goals, namely divergent thinking and idea generation. The most important guiding principle of these techniques is to defer judgement. Other principles are that quantity leads to quality, the crazier the better, and the merge of ideas beget progress (Marakas, 2003). According to Turban et al. (2005), associations form idea chains, i.e. one idea triggers another. Probably, the most famous free association technique is brainstorming. This technique was invented by Alex Osborn (Connolly, 1993). Furthermore, there are multiple commercial products available to support idea generation and brainstorming, for example, Idea Fisher¹, Creative WhackPack², and ParaMind³. It is possible to include such techniques in an REDSS.

For example, in a meeting between RE decision-makers and important stakeholders, brainstorming can be used to generate alternative ideas for the next

---

¹ [catcode.com/ideafisher.html](http://catcode.com/ideafisher.html)

² [www.creativethink.com/](http://www.creativethink.com/)

³ [www.paramind.net/](http://www.paramind.net/)
generation of the system. REDSS can provide additional support for storing, displaying, and refining the ideas.

*Structured relationships* techniques consist of a process in which two or more objects, concepts, ideas, or products are compelled together to create something new. This way, ideas are creatively generated. An example of such a technique is Osborn’s idea checklist, in which the decision-maker is expected to change the decision context into a range of new perspectives. Another example is the Morphological forced connections. This technique stresses the morphological (science of changes in living things) attributes of design problems. The decision-maker is first obliged to write down the attributes of a problem, and then list the alternative options for each attribute. As many options as possible should be identified. The alternatives are subsequently combined and permuted (Marakas, 2003). Marakas argues that especially the Morphological forced connections technique can be easily incorporated in a DSS. Thus, it would also be possible to provide such creative support via an REDSS.

*Group techniques* are developed particularly to promote creativity in a multi-participant context. The guiding principle is to improve group interaction (Marakas, 2003). Two examples of such techniques are the Nominal group technique and the Delphi technique. The nominal group technique is a structured group meeting in which the participants first independently write down their ideas. These are then discussed, and finally individual voting on the ideas is conducted (Delbecq et al., 1975). The Delphi technique is a variant of the nominal group (Jewell and Reitz, 1981). Since both techniques can, according to Marakas (2003), be delivered by a DSS, it should be possible to embrace both in an REDSS.

Thus, if creativity enhancing techniques are integrated in the REDSS, then we can increase the possibility of generating innovative ideas and solutions. The RE decision-maker also gets help storing, displaying, and refining the ideas.

The RE decision processes are knowledge-intensive and knowledgeable RE decision-makers are fundamental for effective and efficient RE decision-making. Consequently, the REDSS should support knowledge sharing and knowledge transfer.

### 9.3.4 Support knowledge sharing and transfer

The REDSS should support knowledge sharing and transfer. Knowledge sharing and knowledge transfer are two related concepts. Knowledge sharing is defined as “the willful explication of one person’s ideas, insights, solutions, experiences (i.e. knowledge) to another individual either via an intermediary, such as a computer-based system, or directly” (Turban, et al., 2007, p 489). Knowledge transfer takes place when “knowledge is diffused from one entity (e.g. an individual, group, or organization) to other entities” (Joshi et al., 2007, p 322). Knowledge transfer is facilitated by knowledge sharing, i.e., someone puts some effort into “giving away” knowledge, although knowledge can be transferred without any particular
knowledge sharing. For example, a discussion can implicitly transfer knowledge from one person to another, without the explicit purpose of sharing knowledge.

In the empirical study of RE decision situations we have seen the importance of knowledge and the problems of not having enough knowledge (see Quotation 136 – Quotation 138). A difficulty in the decision process of establishing requirements which is of vital importance is understanding the problem domain (see section 7.1.3). It is also important to have product or system knowledge (see section 8.6), otherwise the possibilities of making well-grounded system-related decision are reduced. In order to make process-related decisions, the RE decision-maker not only needs knowledge of the RE process (see section 7.1 and section 8.6), but also the systems engineering process.

Quotation 136: “In some way, it has to be possible to perform the job without having worked for so long. You must have a chance from the beginning. There must be support.” (Requirements engineer)

Quotation 137: “With so many persons quitting, the knowledge isn’t here.” (Focus group participant)

Quotation 138: “… when you have a high turnover of people and there are many new people and things like that. And to keep up the quality of it is very difficult, so to speak.” (Requirements engineer)

Thus, the support of knowledge sharing and knowledge transfer is important for supporting RE decision-making. Similar arguments can be found in literature. “An individual’s problem solving and decision making capability is limited by the knowledge available. Having knowledge available to decision makers is crucial to improving individual and organizational performance” (Kim et al., 2004, p 2). In project-based organisations, effective sharing of knowledge across projects can reduce the costs of inventing the same solutions or making the same mistakes several times (Boh, 2007). Support for knowledge sharing and knowledge transfer within information systems engineering teams is important (Joshi et al., 2007). Since RE is an inherent part of systems engineering and since systems engineering is often performed in a project, the same is true for RE. Knowledge sharing and transfer is crucial for high quality RE decision-making.

In order to support knowledge sharing and transfer, we advocate the following guiding principle:

- Apply knowledge management approaches

**Apply knowledge management approaches**

The purpose of knowledge management is to make the sharing of knowledge possible beyond the sharing of information (Alter, 1999). Knowledge management is defined as “the practice of selectively applying knowledge from previous experiences of decision making to current and future decision making activities with the express purpose of improving the organization’s effectiveness” (Jennex, 2005, p
iv). This means that knowledge has to be created, captured, stored, organised, maintained, and delivered in a meaningful form within an organisation (Alavi & Leidner, 1999; Turban et al., 2007). Knowledge can be viewed as contextual, relevant, and actionable information. It gives meaning to data and information (Turban et al., 2007). Knowledge can be explicit or tacit. Explicit knowledge is knowledge that can be easily verbalised in, e.g., words or numbers. Tacit knowledge is much harder to express, since it consists of, e.g., know-how, insights, and skill sets (Polanyi, 1983; Turban, et al., 2007). While knowledge management can be conducted without information technology (IT), when it is IT-based it is called a knowledge management system (Turban et al., 2007).

By applying knowledge management approaches in the REDSS, knowledge sharing and knowledge transfer among RE decision-makers, as well as to and from RE decision-makers and other stakeholders, can be supported beyond the mere sharing of explicit information.

For example, if a field study has been conducted in an user environment with the purpose of eliciting requirements for the interaction between the user and the system, then the person who visited the user environment gains tacit knowledge of the user needs that are difficult to describe explicitly. However, this tacit knowledge can be of vital importance for formulating and deciding upon requirements. So, if the person that has conducted field studies is not the RE decision-maker that needs the knowledge, then the knowledge has to be shared and transferred to him or her with the help of the REDSS.

The potentials of integrating knowledge management in software engineering practices and software engineering decision support have already been identified by Ruhe (2003b). The idea of knowledge management for requirements engineering has also been put forth by Herlea et al. (1997).

**Some available techniques**

Turban et al. (2007) present two main approaches to knowledge management, although often a hybrid approach is used:

- The process approach
- The practice approach

The *process approach* focuses on explicit knowledge where organisational knowledge is codified, e.g., through policies and procedures. This approach is particularly beneficial for organisations with a standardised set of products and services.

For example, RE decision-makers in a customer organisation conduct RE activities when selecting suitable COTS products. The REDSS can give process support in terms of prescribing procedures that have been shown to be feasible in a similar kind of decision process previously. This way, the knowledge from preceding processes is transferred from those persons to the current RE decision-makers.
The practice approach is directed to tacit knowledge, which is primarily transferred through person-to-person contact. Since tacit knowledge is difficult to store, the system support can be links to experts to contact (Turban et al., 2007). Tacit knowledge can also be stored in the form of best practices, which can be captured, stored, and disseminated within the organisation (Choudrie & Selamat, 2005). The organisational knowledge can be stored in a knowledge repository, which should not be confused with a knowledge base used in expert systems. They have different purposes and different mechanisms (Turban et al., 2007). In order to support knowledge transfer of how to effectively and efficiently carry out RE tasks, a method chunks repository can be integrated in the REDSS. This repository consists of reusable parts of methods as potential method building blocks (Mirbel & Ralytė, 2006).

For example, information chunks in requirements documents that are stored in the REDSS can carry links to persons that can be contacted. That person should be an expert, e.g., a user interface design expert or a target domain expert, or the person that has created the information, e.g., formulated scenarios. This way, it will be easier for the RE decision-maker to find the “right” person who can share his or her knowledge with the RE decision-maker.

By applying knowledge management approaches in the REDSS, the possibility for the RE decision-maker to have relevant decision knowledge available increases. To be knowledgeable is essential for decisions concerning both the system and systems engineering process. Appropriate knowledge can make RE decision-making more efficient and more effective, e.g., in idea evaluation and problem solving.

### 9.3.5 Support idea evaluation and problem solving

The REDSS should support idea evaluation and problem solving. According to Miner (1992, p 204), idea evaluation takes place when “a choice must be made among alternative creative proposals or when the appropriate solution to a problem must be identified”. A problem is a situation where a person’s goal does not agree with the current state and where the person does not know how to reach the goal. Problem solving is the search for a path that leads from the current state to the goal state (Parkin, 2000; Reisberg, 2006). Problems need ideas in order to be solved. The ideas need to be evaluated in order to identify which ideas actually solve the problem at hand, and to conclude which idea is most feasible. Thus, idea evaluation and problem solving are not the same, although they are highly related.

The case study showed that the process of managing requirements changes is more of a routine process compared to establishing requirements (see section 7.1.3). The stimuli that initiate the process of requirements changes are, e.g., requirements change proposals and requirements errors. Requirements change proposals can be viewed as ideas that need to be evaluated in order to decide on their acceptance or rejection. Some requirements errors may be obvious and cause a quick fix, e.g., spelling errors. Other requirements errors are much more difficult and call for problem solving, e.g., suspicions that the requirements are based on incorrect assumptions of user characteristics.
The use of a DSS can augment the ability of a decision-maker to tackle large-scale, time-consuming, complex problems (Marakas, 2003). In the same way, an REDSS can amplify the RE decision-makers’ capacity to solve problems and evaluate ideas. For instance, as argued by Power (2002), the first step in a decision process is to define the problem. It is important to define problems well, because if the wrong problem is defined it is not possible to make a correct decision. Organisations are often complex, which makes it harder to recognise a real problem and define it well. Furthermore, it can be difficult to separate real problems from problem symptoms (Power, 2002).

In order to support idea evaluation and problem solving of RE decision-makers two guiding principles are suggested:

- Integrate evaluation approaches
- Integrate a cognitive tool to augment human problem solving

**Integrate evaluation approaches**

By integrating evaluation approaches in the REDSS the evaluation of alternative ideas, e.g., requirements change proposals, are facilitated, so that an appropriate solution can be decided upon.

*Some available techniques*

Turban et al. (2007) suggest four evaluation approaches.

- Multiple goals
- Sensitivity analysis
- What-if analysis
- Goal seeking

Often, the decision-maker has to take multiple goals into account simultaneously. In order to quantitatively compare the effectiveness of suggested alternatives, it is often necessary to transform multiple-goal problem into a single measure of effectiveness. The reason for this is that many models of decision theory are based on the comparison of a single measure of effectiveness. An example of a technique that can be used for handling a manifold of goals is based on utility theory (Turban et al., 2007). Utility theory is a theory of human decision-making which proposes that humans endeavour to maximise utility (Reisberg, 2006). Reisberg explains that the theory consists of three substantial parts; a) the values of humans, b) the goals of humans, and c) the consequences of the choice possibilities. Human values concern both positive aspects valued by the individual and negative aspects the person wants to avoid. Each individual also has short-term and long-term goals that he or she strives for. Each choice possibility facing the decision-maker has consequences in term of costs and benefits, i.e., “things” that take the decision-maker closer to or further away from the goal. The utility theory declares that the decision-maker chooses the alternative with the optimal balance between costs and benefits, and thus gives the optimal expected utility (Reisberg, 2006).
For example, when the RE decision-maker is to prioritise requirements in market-driven software engineering, there are several goals that need to be taken into account, e.g., low implementation cost, high quality, and high competitiveness. With the help of the REDSS, the RE decision-maker tries to balance these goals to find optimal expected utility.

The purpose of sensitivity analysis is to evaluate the effects of a change in the input data or parameters of the proposals. Such an analysis investigates relationships, e.g., the effect of uncertainty in estimating external variables, the robustness of decisions under changing conditions, or the impact of different interactions between dependent variables. Sensitivity analyses can be carried out automatically or through trial-and-error (Turban et al., 2007).

For example, REDSS can support sensitivity analysis of requirements change proposals, such as changing a parameter concerning capacity of some kind. Then, the RE decision-maker, with help of the REDSS, can identify which requirements are affected by the capacity parameter change.

Trial-and-error analysis has two approaches; what-if analysis and goal seeking. The purpose of a what-if analysis is to find out what will happen to the suggested solution if some aspect changes, such as an input variable, an assumption, or a parameter value. Goal seeking is a backward solution approach, in which the values of the inputs required are calculated in order to accomplish a preferred level of a goal. What-if analysis and goal seeking can be carried out with the help of software with modelling possibilities, e.g., Excel (Turban et al., 2007).

For example, if a goal level requirement is proposed to change, the RE decision-maker can do a goal seeking with the help of the REDSS to identify which lower level requirements are affected by the change proposal. Based on that information, an estimation of the cost of changing the goal level requirement can be made.

Thus, by integrating evaluation approaches in the REDSS existing alternatives can be assessed. However, sometimes the RE decision-maker does not have alternative solutions available. Instead, he or she faces an RE problem that needs to be solved in order to reach a decision. Problem solving is not always straightforward, and the problem solving capacity of the RE decision-maker needs to be augmented.

Integrate cognitive tools to augment human problem solving

Cognitive tools should be integrated in the REDSS to augment human problem solving of RE decision-makers. A cognitive tool is any technology “that engages and facilitates specific cognitive activities” (Jonassen, 2003, p 372). Jonassen claims that the key to problem solving is to satisfactorily represent the problem. A cognitive tool can be used for externalising problem representation so that a proper mental model (internal representation) of the problem can be created. A proper mental model can, for example, guide interpretation of information about the problem (Jonassen, 2003). External representation can reduce the complexity of the problem as well as the cognitive load. Problems can be represented externally in a number of ways. An
effective representation in terms of enhancing problem solving is knowledge maps. Other terms for knowledge maps are, e.g., concept maps, semantic networks, or cluster maps. A knowledge map is a graphical representation in which concepts are linked to other concepts (Lee & Nelson, 2005).

The integration of a cognitive tool can support the RE decision-makers, for example, in the design routine of the decision process concerned with managing requirements changes. In this routine, the RE decision-makers solve error reports, as well as change and add requirements. Without finding a solution for the problems at hand, it is not possible to make proper decisions.

For example, if the RE decision-maker investigates the problems of requirements error reports, he or she can benefit from manipulating external representations of the problem at hand. If the REDSS provides a possibility to draw knowledge maps of concepts relevant for the problem, e.g., costs, expected customer satisfaction, and possible work around, then it can be easier for the RE decision-maker to reason about these concepts. Using the map can facilitate the RE decision-maker finding a feasible solution.

Some available techniques

Based on the theory of enhancing problem solving through external representation specifically through knowledge maps, Lee and Nelson (2005) suggest five design principles for effective problem solving performance. The principles are primarily motivated from a learner’s perspective, and not specifically from a decision-maker’s perspective. However, learning can be argued to be not only a necessary part of RE in general, but also in RE decision-making. Thus, Lee and Nelson’s (2005) design principles have possibilities to improve RE decision-making. Lee and Nelson have, based on the principles, developed a tool with four main parts: interface, templates, a knowledge base, and a user-driven database. The design principles are:

- Combinational representation
- Contextual enhancement
- Spatial flexibility
- Property association
- Multiple representations

The first principle, combinational representation, prescribes that the conceptual and corresponding procedural knowledge should be represented together instead of represented as separate artefacts. Conceptual and procedural knowledge have an effect on general mechanisms of problem solving. From a cognitive load perspective, the combination of representations increases a person’s efficiency (Lee & Nelson, 2005). Thus, how and what information should be represented together in the REDSS.

For example, the REDSS displays the requirements of interest together with related task descriptions.
The contextual enhancement principle recommends that the problem context should be described so that the context of the problem solver is reflected, instead of decontextualising the problem from the problem situation. This means that the person is given cognitive control, i.e., able to monitor his or her thinking process while solving the problem. A concrete way of doing this is making it possible to attach commentaries, i.e. descriptions of the relations between the case and the concepts. The explanatory commentaries should not only function as a connection, but also reveal the concepts and processes from an individual perspective, for instance, prior knowledge and personal experiences (Lee & Nelson, 2005).

For example, if the RE decision-maker works with a knowledge map when investigating a problem in a requirements error report, he or she can attach commentaries concerning, e.g., expected customer satisfaction or possible workarounds.

The principle spatial flexibility expresses that the physical space of the medium should not restrict the number of represented concepts. It is more likely that problems are solved effectively when the problem solver can represent their concepts as fully and numerously as they prefer. The conventional ways of fulfilling this principle is by using scroll bars, zooming functions, parent-child relationships. However, there are essential drawbacks to this, not the least of which is overview difficulties (Lee & Nelson, 2005). Instead, Lee and Nelson put forward the use of \( n \)-dimensional types of knowledge maps as an alternative to two-dimensional ones.

The principle called property association advises that the concepts in a knowledge map should be classified, based on the relative importance of association between a concept and a process. This means that the more properties shared between a concept and the process, the more the relative importance of association is increased. This classification accelerates the problem solvers access to relevant information and mental problem representation. The principle property association shares the underlying assumption with object-oriented programming (Lee & Nelson, 2005).

For example, if the knowledge map in the REDSS shows that the requirements at hand share many properties with the test process, then the RE decision-maker can quickly identify that test information is relevant in solving this problem since they share properties in the current problem case.

The fifth principle, multiple representations, states that multiple modes of information should be used for the representation of concepts in knowledge maps. Multiple modes enhance problem solving performance more than when the form of representation is constrained to one mode. The performance of a problem solver depends on how effectively the person constructs mental representations of the problem and how mental representations are supported by multiple external representations. Examples of modes of information that can be used are graphics, sounds, and animation (Lee & Nelson, 2005).

To put it in the words of Lee and Nelson (2005, p 15), the cognitive tool is expected to enhance problem solving performance by allowing the problem solvers to “combine
the concept and processes, present the contextuality of the problem, go beyond the limitations of the 2-dimensional representational space and facilitate external representations of multiple internal representations while solving problems”. The heart of enhancing problem solving performance is to provide effective external representations.

Thus, by integrating cognitive tools and evaluation techniques in an REDSS, the problem solving and idea evaluation of RE decision-makers can be augmented. This is particularly important in the decision process of managing requirements changes. The two routines in the decision process model of Mintzberg’s et al. (1976) are the diagnosis routine in the identification phase and the design routine of the development phase. The diagnosis routine is when the decision-maker is trying to understand the stimuli, e.g., a problem. The design routine concerns developing custom-made solutions or modifying ready-made ones.

Decision-making in RE most often embraces multiple actors. The RE decision-maker needs to communicate with other decision-makers as well as other stakeholders in the systems engineering process. Since several persons are involved in activities of the RE decision processes, coordination to achieve smooth processes is inevitable. This social context of RE decision-makers also generates desirable characteristics of an REDSS.

9.4 The RE decision-maker in the social context

We have identified two REDSS characteristics stemming from the social context in which the RE decision-makers carry out decision-making activities. The social environment is a natural part of RE decision situations that also need to be supported. While supporting single users is complex the social dimension adds more complexity to the REDSS.

We have derived two characteristics from the view of the RE decision-maker in the social context.

- Support decision communication
- Support coordination

9.4.1 Support decision communication

Decision communication should be supported during the whole RE decision process. Mintzberg et al. (1976, p 261) view decision communication as the “active stream of communication throughout the decision process”. Decision communication activities are carried out during the RE decision processes; communication with stakeholders, dissemination of decisions, and negotiation (see section 7.1). We have also identified decision communication problems, which argue have a potential negative effect on RE decision quality (see section 8.2). Decision communication is important both in the pre-choice as well as the post-choice phase of RE decision processes. Communication is vital in order to obtain relevant information as well as opinions
from stakeholders or other experts. This enables the RE decision-maker to make more well-informed decisions. Communication is also important in order to reach an agreement concerning the decision at hand, otherwise the implementation can be more difficult and the benefits of the decision can be reduced. In the post-choice phase, communication is essential so that the persons who are going to implement the decision, e.g., a requirements change, are aware of the existence of the decision.

We suggest two guiding principles to support RE decision communication:

- Provide additional communication paths
- Provide negotiation facilities

**Provide additional communication paths**

The REDSS should provide additional communication paths, i.e. make it possible for the RE decision-maker to communicate with stakeholders and experts in other than just the traditional ways of face-to-face or via phone.

In RE decision-making, decision communication with stakeholders frequently occurs. Often face-to-face communication or communication via the telephone is effective and efficient in decision-making. However, sometimes such communication is difficult or expensive, e.g., when multiple requirements stakeholders at different locations communicate and where several artefacts are also involved. In such cases, computer-mediated communication can be a feasible alternative. The primary function of computer-mediated communication is to support direct communication between several persons (Dix et al., 2004).

For example, if the RE decision-maker has to discuss the screen layout with a customer located in another part of the country, he or she can have a conference via the REDSS where both parts can view and change the layout while discussing it.

The REDSS should also facilitate dissemination of decision outcomes to the stakeholders that are going to act upon it. The excellence of a decision does not matter if these persons who are going to implement it are unaware of its existence. To avoid unawareness, a notification mechanism (Dix et al., 2004) can be implemented in the REDSS. This way, the relevant stakeholders become informed of new decisions.

For example, if a requirement has been changed, the REDSS automatically notifies these stakeholders who should be informed of the change.

Thus, by providing additional communication paths decision communication, RE decision communication can be supported.

**Some available techniques**

A DSS where communication technologies are central is called a communication-driven DSS (Power, 2002). Communication is an important part of cooperation. According to Grudin (1994), co-working features should be integrated in the tool or system that the user frequently uses at work. If co-working features are in a separate
application, the work flow of the user is reduced, and consequently the co-working features will be of less use (Grudin, 1994). Thus, the co-working features concerning decision communication should be integrated in the REDSS.

There are many tools for computer-mediated communication available on the market. Such tools provide indirect support of decision-making (Turban et al., 2007). Examples of computer-mediated communication tools are provided by Dix et al. (2004), Benyon et al. (2005) and Turban et al. (2007) and include chat systems, interactive whiteboards, bulletin boards, shared information spaces, and virtual meeting systems. Chat systems provide real-time text conferencing. An interactive whiteboard is used in the same way as a physical whiteboard, but is shared on a computer-based drawing surface. In a bulletin board, messages are distributed to particular newsgroups. A shared information space consists of information and additional features needed for cooperative work. Virtual meeting systems enable meetings between participants located at different places (Dix et al., 2004; Benyon et al., 2005; Turban et al., 2007).

For example, when the RE decision-makers discuss ideas and solutions with those responsible for the entire system and the persons who are responsible for subsystem requirements, it can be necessary to document the discussions so that everyone involved in it will be remembered, e.g., the pluses and the minuses of the ideas and solutions. If, for example, an interactive whiteboard is provided by the REDSS, the notes taken on the whiteboard can easily be stored and retrieved together with, for instance, the requirement that were discussed.

The way decision communication with stakeholders should be carried out differs from situation to situation. In some situations and with some stakeholders face-to-face communication is preferable, e.g., critical requirements discussions with customers. In other situations and with other stakeholders, computer-mediated communication can be the most efficient, e.g., collecting opinions from geographically distributed experts. The more communication paths that are available, the more the possibility finding an effective decision communication for every situation increases.

**Provide negotiation facilities**

The REDSS should provide negotiation facilities. Some RE decisions are negotiated, e.g., if a requirement change proposal is going to be approved or not; and which system requirements belong to which subsystem (see section 7.2).

**Some available techniques**

Negotiated decisions can be supported by using a group decision support system (GDSS). A GDSS facilitates the performance of group decision-making tasks, by providing, e.g., an opinion meter, electronic brainstorming, or voting tools (Power, 2002). A particular type of GDSS is negotiation support systems (Holsapple et al., 1996). Negotiation support systems support bargaining, consensus seeking, and conflict resolution, and their purpose is to achieve optimal settlements among negotiators (Bui et al., 1992; Lim, 2003). A specialised negotiation support system for
requirements negotiation has been developed by Ruhe et al. (2002). It is called Quantitative WinWin and focuses on requirements selection, based on three aspects: a) the stakeholders’ preferences, b) the business value of requirements, and c) a given upper limit development effort.

However, Baltes et al. (2005, p 175) stress that we should “exercise significant caution when investigating and adopting computer-mediated means for group decision making. Computer-mediated communication may be an efficient and rapid means of disseminating information, but research to date suggests that it is not the most effective means of making group decisions.” Thus, there is the potentiality of supporting decision communication via an REDSS, although there can also be serious drawbacks that need to be taken into account.

The social context of the RE decision-maker also calls for coordination of the multiple actors within the RE decision situations.

9.4.2 Support coordination

The REDSS should support coordination. “Coordination is managing dependencies between activities” (Malone & Crowston, 1994, p 90). There are several different types of coordination processes, e.g., the management of shared resources, producer/consumer relationships, simultaneous constraints, or management of task/subtask dependencies (Malone & Crowston, 1994). Since there are several persons involved in the RE decision process, there is a need for coordination. In the case of RE decision situations, we have identified coordination problems about the coordination of the way of working and time-consuming coordination with stakeholders. These problems affect the RE decision-makers and hence there can be potential quality problems in the RE decision process (see section 8.2). In addition, some decision matters in the RE decision process concern coordination (see section 7.2). Therefore, coordination should be supported by an REDSS, so that the decisions concerning RE coordination are improved, i.e., increased effectiveness, and reducing the time needed for coordinating activities, i.e., increased efficiency.

To support coordination, we suggest the following guiding principle:

- Integrate coordination technologies

Integrate coordination technologies

Coordination technologies should be integrated in the REDSS. These technologies support the management of interdependencies between activities (Benyon et al., 2005). By integrating such technologies, the REDSS can facilitate decision-making with the purpose of harmonising RE activities, which means more effective work-related RE decisions. However, the integration of coordination technologies can also make RE decision-making more efficient since they can reduce the time-consuming coordination with stakeholders.

For example, the REDSS can provide guidance in the way of working. The RE decision-maker can be given advice about how to write requirements, in term of
styles, level of details, etc., and warn when there is a divergence. This can produce better cohesion among the requirements, which in turn makes the requirements specification less muddled and easier to use. In addition, this makes it easier to correctly interpret the requirements decisions.

Some available techniques

Benyon et al. (2005) describe that management of interdependencies between activities can embrace:

- Specification behaviours
- Planning behaviours
- Scheduling behaviours

Specification behaviours concern, for example, creation of shared goals. An example of planning behaviour is making the set and order of tasks agree. Scheduling behaviours are, for instance, the assignment of tasks to individuals or groups. The coordination technologies support the structuring and organising of collaborative work, where the participant can be distributed in space and time (Benyon et al., 2005).

For example, the REDSS can facilitate scheduling meetings with the stakeholders that the RE decision-maker needs to discuss or negotiate RE decisions with.

Examples of commonly used coordination technology include shared calendars and workflow systems (Benyon et al., 2005). Shared calendars facilitate scheduling and coordination among several persons (Benyon et al., 2005). A workflow technology is “any technology designed to (in some way) give order to or record the unfolding of work activity over time by, for example, providing tools and information to users at appropriate moments or enabling them to overview the work process they are part of or to design work processes for themselves or others or whatever” (Bowers et al., 1995, p 51).

For example, the REDSS can enable the RE decision-makers to overview the decision processes. The REDSS can show what decision-making activities have already been conducted and what the upcoming planned activities are. This can facilitate for the RE decision-maker in judging if the planned activities need to be adjusted or if the plan of RE decision-making activities is still appropriate.

Thus, by supporting coordination via coordination technologies in the REDSS, RE decision-making can be improved in different ways. It can afford support in order to make informed work-related RE decisions, to harmonise ways of working, and facilitate collaborative work.

Based on RE decision-making in the social context two desirable characteristics have been described. These, together with the characteristics stemming from the nature of RE decision-making tasks and the needs of the RE decision-maker as a single user, form a visionary view of how a requirements engineering decision support system (REDSS) can be constituted.
9.5 Chapter summary and reflections

In this chapter, we explore the scope of REDSS with a visionary purpose. We identify nine empirically grounded desirable characteristics of an REDSS that are based on the needs of RE decision-makers and the nature of RE decision-making. This means that we focus on what is generic, and not on specific RE tasks. For each characteristic we suggest one or more guiding principles that direct further efforts concerning how to find a solution that can fulfil the characteristic. The guiding principles are empirically and theoretically grounded. The characteristics and their guiding principles are summarised in Table 3. For all guiding principles we also present some available techniques, which are theoretically grounded.

Table 3, Desirable characteristics of an REDSS and their guiding principles

<table>
<thead>
<tr>
<th>Desirable characteristics</th>
<th>Guiding principles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduce the cognitive load</td>
<td>Present both overview and details</td>
</tr>
<tr>
<td></td>
<td>Provide memory aid</td>
</tr>
<tr>
<td>Ensure high usability</td>
<td>Follow usability design principles</td>
</tr>
<tr>
<td>Support availability of different types of information</td>
<td>Apply information visualisation on different levels of use</td>
</tr>
<tr>
<td></td>
<td>Combine data-driven and document-driven DSS techniques</td>
</tr>
<tr>
<td>Support different types of decision matters</td>
<td>Integrate requirements decision support techniques</td>
</tr>
<tr>
<td></td>
<td>Integrate RE process decision support techniques</td>
</tr>
<tr>
<td></td>
<td>Integrate requirements-based decision support techniques</td>
</tr>
<tr>
<td>Support creativity and idea generation</td>
<td>Integrate creativity enhancing techniques</td>
</tr>
<tr>
<td>Support knowledge sharing and transfer</td>
<td>Apply knowledge management approaches</td>
</tr>
<tr>
<td>Support idea evaluation and problem solving</td>
<td>Integrate evaluation approaches</td>
</tr>
<tr>
<td></td>
<td>Integrate cognitive tools to augment human problem solving</td>
</tr>
<tr>
<td>Support decision communication</td>
<td>Provide additional communication paths</td>
</tr>
<tr>
<td></td>
<td>Provide negotiation facilities</td>
</tr>
<tr>
<td>Support coordination</td>
<td>Integrate coordination technologies</td>
</tr>
</tbody>
</table>

The desirable characteristics and guiding principles have many potential benefits for RE decision-making. If the RE decision-maker experiences a supportive, smooth and
pleasant tool, then the chances of frequent and appropriate use of the REDSS increase. If basic users' needs are ignored, then the REDSS will be less valuable and too cumbersome to use. Since the RE decision-maker conducts several different tasks it is important for the REDSS to be highly helpful and valuable that the system provides support for these actual decision-making tasks. Otherwise, the REDSS will not be suited for its purpose. The social environment is a natural part of RE decision situation that also needs to be supported.

A lowered cognitive load enables the RE decision-maker to concentrate on the decision activity at hand. By presenting both overview and details, the difficulties RE decision-makers have when obtaining a general view can be reduced. A general overview facilitates the understanding, because by contextualising the details of requirements information it is easier for the RE decision-maker to properly interpret the information. Since recall of information from the human memory requires mental exertion and therefore affects the cognitive load, providing a memory aid can reduce the unnecessary cognitive load.

If high usability is ensured, then the possibility that the REDSS actually is used increase. High usability extend the chance of that the system’s decision-supporting capabilities are correctly and efficiently utilised. It also reduces the cognitive load, since the decision-maker can use the conscious information process capabilities to carry out the decision-making activities and not to manoeuvre the REDSS.

By making relevant information available to the RE decision-makers, RE decision-making is expected to be more effective and efficient. Availability of information means not only access to relevant information, but also mental availability. The information has to be smoothly and correctly interpreted and understood by the RE decision-makers. Thus, the needed information should be possible to reach with the help of REDSS and the information should be properly presented. If the mental availability of information is low, there is a risk the information will be provided in an unsuitable way making it unnecessarily difficult for the RE decision-maker to find and make sense of.

The REDSS should not just support the main decision matters but also the sub decisions, since the sub decisions affect the main decisions as well as the development process. Furthermore, it should support both system-related and work-related decisions, since different kinds of decision matters require different decision support techniques. By supporting decisions concerning the requirements as such, the effectiveness of system-related decision can improve, and also increase the efficiency of RE decision-making tasks concerning requirements. By supporting RE decision-makers in determining suitable ways to carry out the RE process, positive effects on work-related decisions can be gained and the RE process can be more efficient. Furthermore, using the “right” RE techniques increases the potentiality of making more informed system-related decisions. By supporting decisions that use the existing requirements as input, we can get an extension of the REDSS and bonus effects on decisions made in the systems engineering process not concerned with requirements as such, or the RE process. The more RE decision support techniques
that are included in an REDSS, the higher the decision-supporting capacity of the system, since more different kinds of RE decision matters are supported.

The support of creativity and idea generation increases the possibility of generating innovative ideas and solutions. The RE decision-maker is also helped to store, display, and refine ideas.

The sharing and transfer of knowledge concerning the application domain, the system, and RE practices increases the possibility of the RE decision-maker to have relevant decision knowledge available. The RE decision processes are knowledge-intensive and knowledgeable RE decision-makers are fundamental for effective and efficient RE decision-making. It is essential to be knowledgeable of decisions concerning both the system and systems engineering process. Appropriate knowledge can make RE decision-making more efficient and more effective.

The integration of evaluation approaches in the REDSS facilitates the evaluations of alternative ideas. Moreover, cognitive tools augment the problem solving capacity of RE decision-makers.

The more that communication paths are made available increases the possibility finding an effective decision communication for every. Decision communication is vital in order to obtain relevant information as well as opinions from stakeholders or other experts, so that the RE decision-maker can make well-informed decisions. Communication is also important for reaching an agreement concerning the decision at hand. Otherwise, the implementation can be more difficult and the benefits of the decision can be reduced. In the post-choice phase, communication is essential so that the persons to implement the decision are aware of its existence.

The support of coordination improves decisions concerning RE coordination, i.e., increased effectiveness, and reduces the amount of time needed for coordinating activities, i.e., increased efficiency. It enables informed work-related RE decisions, harmonises ways of working, and facilitates collaborative work.

Thus, there are multiple ways of improving RE decision-making and there are potentials by implementing the characteristics. However, it is easy to speak glibly of this. In practice, there are several challenges and obstacles to take into account, not the least of which is the economical perspective. We do not claim it is realistic to just implement all the desirable characteristics and guiding principles. Instead, our contribution is to provide inspiration for RE tool developers that is based on empirical research findings. Our contribution also directs research efforts of RE decision support researchers, since, as stated by Ngo-The and Ruhe (2005), research within the field of RE decision-making and RE decision support is in its infancy.

One of the interviewees in the case study put forth ideas that are worth thinking about (see Quotation 139 and Quotation 140).

Quotation 139: “It is very easy to think that once we have a tool, it will be settled. However, you have to know what you want to do before you acquire a tool, so to
speak. [...] you have to see how a tool should support us in the work. [...] Since it was like, well, here is a database. Do something good with it. People weren’t able to do that, so to speak. [...] You have to stay alert so that you don’t create an administrative monster that needs to be fed with numbers every morning. [...] However, before you do that you must know… can we manage to maintain this or will this just be flavour of the week, so to speak, to put in and make requirements traces for example. Will we keep this updated or will it be just once and then never again.” (Requirements engineer)

Quotation 140: “You cannot put in every fun thing [into the tool] just because it is possible.” (Requirements engineer)

The RE process consists of decisions and decision-making activities that are critical for both the system to be as well as the systems engineering process. The RE decision-maker’s abilities and capabilities can be enhanced if appropriate RE decision support is provided. RE decision support system characteristics have the potential to suggest and direct research and development efforts concerning decision support to RE decision-makers.
10 Discussion

In this chapter, we conclude the thesis by reviewing and discussing the results. Firstly, we discuss the contributions, then, we present some general reflections concerning them, and finally we outline future work.

10.1 Contributions

Requirements engineering (RE) is an important part of systems engineering and thus the quality of RE decision-making affects the quality of the developed system as well as the efficiency of the systems engineering process. Our aim is to contribute to the improvement of RE decision-making by providing RE decision support that augments the human RE decision-maker’s possibilities of carrying out decision-making activities in the whole RE decision process. Many challenges remains before this aim can be fulfilled. However, as a step towards this aim this thesis addresses two such challenges. These concern the lack of understanding of RE decision-making in practice (especially from a holistic and human-centred perspective) and the limitations of existing RE decision support (that focus on specific RE decision problems on a fairly detailed level). These problems underpin the objectives of the thesis; to provide a portrayal of RE decision-making in practice and to suggest desirable characteristics of an RE decision support system. The purpose of the portrayal and characteristics is to serve as a road map that can direct the efforts of researchers addressing RE decision-making and RE decision support problems. Our intention is to widen the scope and provide new lines of thought about how decision-making in RE can be supported and improved. More precisely, there are three main contributions in this thesis of which the first constitutes the foundation of the other two. These two are the principal contributions.

- A generic decision situation framework
- A portrayal of the decision situation of RE decision-makers in practice
- Desirable characteristics of an RE decision support system

In order to facilitate a systematic investigation of the decision situation of decision-makers, guidance is needed so that all relevant aspects are covered. A generic framework of decision situations can provide such guidance. Furthermore, to the best of our knowledge, no such framework exists. Thus, our first contribution, a generic decision situation framework, is a theoretically based framework that takes a holistic perspective on the decision situation from the viewpoint of a decision-maker. It consists of aspects which form a contextual whole that concern a decision-maker. The decision situation framework can be used to portray a particular decision situation of a certain decision-maker in a human-centred and holistic manner. As a result of its generic nature, it has potentials beyond the domain of RE decision-making and RE decision support. It can be applied to any field where it is relevant to study decision-making, for example, in the development of other types of
information systems. While other information systems also support decision-making, they perhaps do not specifically address decision-making activities.

In order to effectively improve RE decision-making, we have to understand the nature of RE decision-making and the needs of RE decision-makers. This means we have to understand how RE decision-making is conducted in practice. In the research area of RE decision-making, there is no cohesive body of knowledge concerning this matter. To the best of our knowledge, no one has conducted empirical in-depth studies in order to give a detailed account of RE decision-making in practice. Therefore, in order to take a step towards such a body of knowledge, our second contribution is a portrayal of the decision situation of RE decision-makers. We provide a holistic and in-depth description of one information-rich case. The description is human-centred and takes different aspects related to decision-makers in requirements engineering into account. Based on an empirical case study, we describe two different RE decision processes, their decision-making activities and the decision matters they encounter. The characteristics of the decision matters as well as the information sources used by RE decision-makers are also elaborated. A number of factors are described that directly or indirectly influence RE decision-making and, as a consequence, may have an effect on the decision outcome. Related to each factor, we identify difficulties and problems that can cause potential quality problems in RE decision-making.

The research area of RE decision support is relatively immature. As a consequence, the existing RE decision support is limited and there is no consensus in the field of the scope of RE decision support. Therefore, the exploration of possible constitutions of RE decision support is valuable in order to take further actions towards providing effective and efficient support. This exploration resulted in our third contribution, represented by nine empirically grounded desirable characteristics of an RE decision support system (REDSS). The purpose of such an REDSS as well as the characteristics is to augment the human RE decision-maker’s possibilities of carrying out decision-making activities in the whole RE decision process. The characteristics are based on the needs of RE decision-makers and the nature of RE decision-making. This means we focus on generic aspects of RE decision-making activities, and not on specific RE tasks. For each characteristic, we suggest one or more guiding principles that direct further efforts concerning how to find a solution which can fulfil the characteristic. The guiding principles are empirically and theoretically grounded. For all guiding principles we also present some available techniques, which are theoretically grounded.

These contributions can together function as a road map for researchers striving towards building a coherent body of knowledge of RE decision-making; developing tools for RE decision support; as well as in other ways working with RE decision-making improvements. With the help of the road map metaphor, we can argue the “map” consists of many “places” that can be used to identify relevant “starting points” and “destinations”. Moreover, the map shows “roads” that can direct a “journey” from a starting point to a destination. This means that a “traveller”, i.e., persons working with RE decision-making and RE decision support, can choose, for
example, idea generation as a starting point in the RE decision process of establishment of requirements. As a destination for the journey, the traveller can choose, e.g., to focus on integrating creativity enhancing techniques in an RE tool for the purpose of increasing the number of generated ideas, which in turn increase the possibility that some idea or ideas are both novel enough to lead to a new product that is able to be competitive on the commercial market. Furthermore, the traveller can choose to use the road of supporting serendipity so that the RE decision-maker is exposed to related information which can break down rigid thinking and lead to novelties that in turn can bring about effective requirements decisions. The road map that we provide has “white spots”, i.e. it is incomplete and we can not guarantee “safe journeys” along the roads. Nevertheless, to the best of our knowledge, this is the first road map in the relatively unexplored “jungle” of the field of RE decision-making and RE decision support that is based on empirical findings with a holistic and human-centred perspective.

The results of this thesis primarily contribute to requirements engineering. However, they also have the potential, to some extent, to contribute to other domains that embrace an interest in studying decision-making and developing decision support. The decision situation framework is by its nature generic and can be used in different domains to structure the study of decision situations. Additionally, the suggested desirable characteristics of an REDSS can also be desirable in decision support systems for other domains. The example mentioned above, i.e., increasing the number of generated ideas by supporting serendipity, is, of course, not unique for RE. However, we can only make claims of the novelty and significance of contributions in the field of RE.

At a general level, the thesis contributes by combining three different, although related, fields. These are requirements engineering, decision support systems (DSS), and human-computer interaction (HCI). RE is the application domain, DSS gives “solutions”, and a user-centred (i.e. human-centred and decision-maker-centred) perspective of HCI offers the lines of thought that permeate the thesis. We have found this way of working to be fruitful. Thus, we recommend other researchers to utilise the knowledge of these three fields in combination in order to address RE decision-making.

In summary, the most important contributions are the enhanced understanding of RE decision-making in practice and the picture of a visionary REDSS that we have painted. They are the results of a research process consisting of theoretical as well as empirical work. The process is reflected upon in the following section.

10.2 Reflections on the research process

The fundamental viewpoint that permeates this thesis is that RE decision-making can be substantially improved by RE decision support systems (REDSS) based on the actual needs of RE decision-makers and the actual generic human decision-making activities that take place in the RE decision processes. This viewpoint has directed the research process. It shows the necessity of understanding human decision-making in
general, as well as decision-making in practice in the current domain. If we do not know what to support then we cannot know how to provide effective and efficient support. Thus, the first steps of the research process were to investigate and learn more about decision-making. First, we conducted a literature analysis that provided the general perspective of human-decision-making. Secondly, we conducted a case study in order to investigate in practice the decision situation of decision-makers in RE. Another essential direction provided by the fundamental viewpoint was that we have to know what should desirably characterise an REDSS. Furthermore, it should be based on actual needs and actual human decision-making activities. Therefore, the third step was to synthesis, meaning the empirical results were merged with relevant theories and techniques.

The strength of the literature analysis is the scope. The literature analysis embraced a variety of theoretical traditions of human decision-making theories and literature from the field of decision support systems. Since several different and related areas were covered, it was possible to create a holistic view of decision-making from the decision-makers’ perspective. Having a holistic perspective denotes trying to view decision-making as a whole, to be understood as a complex system that is greater than the sum of its parts (Patton, 2002). The literature analysis facilitated the identification of the key aspects that are related to a decision-maker in a decision situation as well as obtaining the theoretical foundations of human decision-making that are relevant for decision support. The theoretically grounded identification of key aspects of the decision situation constitutes a firm foundation for the empirical studies.

However, a weakness of the literature analysis, in some sense, is that it is somewhat shallow. Since our purpose was to identify the key aspects of decision situations we had to focus on trying to cover the variety of research traditions of the study of decision-making. Therefore, we could not include all the theories in a certain research tradition. We had to make a trade-off between breadth and depth, due to time limitations, and thus breadth was prioritised in this case, since we have chosen a holistic perspective.

Another weakness of the decision situation framework is that it has only been used by its inventors, and it is possible that we are biased when using it. A further weakness is that the benefits of using the framework for analysis have not been compared in an experimental way. We cannot with certainty claim that the results of using the decision situation framework are positive in comparison to other ways of performing investigations. This clearly requires further research.

In the case study of the decision situation of RE decision-makers, there were both main decisions and sub-decisions among the RE decision matters. Main decision means the last decision that is made in each decision process, and can also be seen as defining what the decision-makers’ work “is all about”. The sub-decisions, however, were not regarded as decisions by the interviewees. Instead, they were hidden as activities, judgements, and trade-offs. Nevertheless, the sub-decisions do have an impact on both the current work and also the outcome of the decision processes. The
sub-decisions may escalate. Escalating commitment means that a person, group, or organisation invests more and more in an initial decision. This can lead to getting stuck in a certain course of action (Miner, 1992; Staw, 1997). For example, the RE decision-makers decide what level of effort a requirements investigation should have, which is also indirectly a prioritisation of requirements. As a consequence, the RE decision-makers invest more work load in some requirements and less in others, which can influence the experienced level of importance of the requirements. Therefore, it is important to also be aware of decisions that are not directly expressed as such.

There are two different RE decision processes: a) establishment of requirements, and b) management of requirements changes. While there are similarities between the processes, more importantly there are differences. Highlighting the differences allows us to examine the diversities and identifying dimensions in the RE decision-making activities. This is elaborated in section 7.1. We assume these two decision processes exist in most, if not all, development projects. However, a weakness is that the case study consisted of only one company, although, additional transferability interviews were carried out with IT consultants in an IT consultancy firm. While the result cannot be generalised, the transferability interviews indicate that the results agree, to a large extent, with the experiences the IT consultants have of other types of organisations; although there are also aspects that differ between companies. Thus, we can conclude that the results of the case study are partially transferable to other organisations.

We do not claim to have a complete set of decision-making activities or decision matters. More research is needed to determine which activities and decision matters occur in most RE decision processes as well as which activities vary.

We identify six factors which directly or indirectly affect the RE decision-maker in the case study. These factors are interrelated. Each factor consists of several problems, which are related to decision-making theories. It is reasonable to assume that the factors are not unique for the company in the case study, and that they can be found, more or less, in every development organisation. The transferability interviews showed that we can conclude that the results of the case study are not exclusive for just this particular organisation.

We can also conclude that when one or more of these problems appear in a development organisation, then there are potential quality problems in the RE decision process. Nevertheless, a weakness of this research is that only one company was studied. Therefore, it is difficult to determine, with certainty, to what extent the factors apply to other development companies.

The result of the synthesis is nine desirable characteristics for a visionary REDSS, which take several different aspects of RE decision situations into account. A major strength of the characteristics is that they are empirically grounded. This grounding makes certain that the suggested characteristics are based on the actual needs of RE decision-makers as well as the actual nature of the RE decision-making activities. An
advantage of focusing on the generic human activities that take place in the RE decision processes is that the generic human activities are independent of the RE context, such as RE maturity level or type of application. As a consequence, the drawback of only using one case for the derivation of characteristics is reduced. RE activities and tasks per se are more context dependent. For example, the characteristic “support idea evaluation and problem solving” can always be valuable independent of development context, whether it concerns in-house or market-driven development.

Further strength is that the guiding principles are empirically as well as theoretically grounded. Thus, the principles are based on the actual needs of RE decision-makers and the actual nature of RE decision-making activities. The theoretical grounding brings additional underpinnings to the suggestion of these guiding principles. Moreover, the theoretical grounding of the available techniques shows there are possible ways of fulfilling the guiding principles and the characteristics. However, a weakness is that the characteristics have not been empirically validated. Therefore, it is not proved that the characteristics have an actual impact on the outcome of RE decision-making or to what extent they have an effect. Consequently, we cannot conclude which characteristics are the most critical for RE decision support. Nonetheless, the empirical and theoretical grounding demonstrate that the characteristics, together with the guiding principles and available techniques, can bring substantial value to RE decision-makers.

For clarification, we do not make any claims concerning economical aspects or other practical issues. Instead, our intention is to explore and widen the scope of how RE decision-making can be supported in order to provide a road map for further research and development.

If we had also chosen to use pure theoretical grounding of characteristics we may have identified more characteristics. By a pure theoretically grounded characteristic we mean a characteristic that can only be found in literature and not in our empirical data. We decided to focus on empirically grounded characteristics, mostly because our fundamental viewpoint is that decision support should be based on actual needs and activities, but also because of time limitations.

However, based on the results of the case study and the literature analysis of decision-making and decision support it is possible to argue that the proposed characteristics, and the guiding principles whose purpose is to direct further efforts concerning how to find a solution, probably positively affect the effectiveness and efficiency of RE decision-making. The set of characteristics can also be viewed as an initial hypothesis to be further developed and validated by future research.

In summary, the main strengths of the research process are a) the firm theoretical foundation that the literature analysis provided, b) the empirical platform that was the result of the case study, and c) the empirical as well as theoretical grounding in the synthesis. The main weakness is that the research process does not consist of all the steps of a full research cycle, since we have not empirically validated the result.
from the synthesis. A full research cycle would have been far too time-consuming. However, we do not consider the lack of empirical validation a large problem, since our research problem is, by nature, explorative and our purpose is to provide a road map for this relatively immature field. Thus, the results can constitute a platform for launching new research projects.

### 10.3 Future work

The purpose of this thesis is to provide a road map that can direct the efforts of researchers addressing RE decision-making and RE decision support problems. The contributions provided widen the scope and give new lines of thought about how decision-making in RE can be supported and improved. Hence, the results of this thesis are not final and conclusive, but instead should be viewed as starting points for future work. Several possibilities for future work based on the results follow:

- Make the decision situation framework more prescriptive
- Study the dimensions along which the decision situation of RE decision-makers differ and what causes the differences
- Scrutinise the details of RE decision situations
- State the cause and effect relationships within an organisation concerning factors and the quality of RE decision-making
- Suggest ways of improving RE decision-making beyond providing computer-based RE decision support
- Build prototypes and empirically validate the characteristics and guiding principles for REDSS
- Suggest effective and efficient concepts for user interface design as well as technical solutions for REDSS features
- Investigate practical aspects, such as cost-benefits, for the implementation and use of the characteristics
- Develop a summative evaluation method that can assess to what extent RE tools have decision-supporting capabilities
- Investigate the decision-supporting capabilities of existing RE tools
- Develop a formative evaluation method that can help identify the decision-supporting problems of an RE tool in order to have an immediate effect on the qualities and features of the tool

By making the generic decision situation framework more prescriptive with regard to how it can be utilised, it can be more supportive. It has the potentiality to be extended to include, for example, suggested activities, guidelines, techniques, or proposed relevant theories. This way, the investigation of decision situations can be facilitated. Additionally, the information can be gained in a more efficient way and it can also be more useful to its stakeholders. Thus, the framework can be developed further. It also needs to be tested in different settings.

A cohesive body of knowledge concerning RE decision situations and all its aspects does not exist. Such a body of knowledge would increase the possibility of making
significant improvements of RE decision-making. There are several parts of the
development of a cohesive body of knowledge that need to be included. It would be
valuable to see along which dimensions the decision situations differ and what causes the
differences. The decision situation of RE decision-makers in different contexts should
be compared, such as market dimension, organisational culture, size of projects, and
maturity of the RE process. For instance, we can study what parts of the RE decision
situation have a low context dependency and what parts have a high context
dependency, i.e., are specific to certain types of projects or organisations, for
example.

The details of RE decision situations need to be scrutinised in order to develop a
cohesive body of knowledge concerning RE decision situations. The advantage of the
viewpoint of the RE decision situation in this thesis is that it is holistic. Consequently,
it includes all or most of the key aspects important for a decision-maker. However, future work should also have a narrower scope so that in-depth studies of
each aspect of the RE decision situations are conducted. One such study can, for
instance, focus on drawing a social map of each decision-making activity or focus on
decision outcome, i.e., how the decision is expressed.

There is a need to state cause and effect relationships within an organisation with regard
to factors and the quality of RE decision-making. We have identified several factors
that we argue have an impact on the quality of RE decision-making. However, the
cause and effect relationships have not been empirically investigated. Such research
would be valuable in order to prioritise the importance of the factors in relation to
each other, for example.

The knowledge of RE decision-making in practice can be used to suggest ways to
improve RE decision-making beyond providing computer-based RE decision support. There
can be other ways to improve the effectiveness and efficiency of RE decision-making,
such as checklists, examples of “best practices”, decision-making techniques, and
knowledge an RE decision-maker should have. One way to work with RE decision-
making improvements is to use prescriptive decision-making theories. These theories
give guidance on how to act within a decision process and they can be used in
different ways. For example, they can be used by a decision-maker as a checklist for
what to do and how to think or they can be a source for learning how to carry out
effective and efficient decision-making.

Future research should include building prototypes that can be used in order to
empirically validate the effectiveness of the characteristics and guiding principles for
REDSS that we have suggested. The characteristics and guiding principles are
empirically as well as theoretical grounded. However, they also need to be tested so
that we can state their actual usefulness. For example, which characteristics afford
the best effects, which characteristics are extra valuable in a certain context, and
which characteristics provides the most significant increase of efficiency. Another
aspect is that our purpose has been to expand the scope of RE decision support and
be visionary. Thus, practical aspects, such as cost-benefit, have not been taken into
account. It is important to consider feasibility aspects in future work. Prototyping can
also be used to investigate and suggest effective and efficient concepts for user interface design as well as technical solutions for REDSS features.

There is also a lack of means that enable systematic evaluation of the decision-supporting capabilities of RE tools. Based on the characteristics and guiding principles, a summative, criteria-based evaluation method can be developed. Such a method can assess to what extent RE tools have decision-supporting capabilities. The purpose of the method could be to explore to what extent RE tools are capable of supporting RE decision-making. The usefulness of carrying out evaluation with the help of such a method is that it can promote better REDSS by enhancing the body of knowledge of the decision-supporting capabilities of RE tools. Thus, learning with regard to the potential of RE tools functioning as an REDSS is supported. Hence, future work should also investigate the decision-supporting capabilities of existing RE tools. The method, as mentioned above, should be summative, which means it is concerned with the intrinsic values of the evaluation object, i.e., the RE tools. Hence, its purpose is not to suggest changes of the tools. Future research can be directed to develop a formative evaluation method that can help to identify the decision-supporting problems of an RE tool in order to have an immediate effect on the qualities and features of the tool. Such a method would make easier for an RE tool developer with the intention of increasing the decision-supporting capabilities of the tool to obtain concrete ideas for improvements.

All the suggested future work mentioned above is relevant. However, the suggestions with the highest priority are the ones that further complement the picture of RE decision-making in practice. A comprehensive understanding of the actual decision situations of RE decision-makers is essential for RE decision support and improvements. Hence, studying the dimensions along which the decision situations of RE decision-makers differ and what causes the differences, as well as scrutinising the details of RE decision situations are of main concern.

In order to illustrate a way to use the characteristics in the future, we have taken a step further and developed an evaluation method called DESCRY. The purpose of method is to investigate to what extent RE tools have decision-supporting capabilities.
11 Taking a step further

In this chapter, we take a step further and provide an illustration and concretisation of how the characteristics can be used in the future. Methodological considerations are presented, and the evaluation method, DESCRY, is introduced. Its criteria and their related questions are also outlined. In addition, we illustrate how DESCRY can be used by applying it on a commercially available RE tool.

RE tools need to be evaluated for different reasons, e.g., a potential RE tool buyer wants to evaluate which tool is the most feasible for them. Thus, they need to identify their own needs and then compare RE tools. There are several evaluation methods for tool selection available, e.g., COSTUME (Carvallo et al., 2005), the R-TEA approach (Matulevičius, 2005), and the value-based tool selection approach (Heindl et al., 2006). Another reason for RE tool evaluation is when RE tool developers want to carry out evaluations to identify improvement needs.

To the best of our knowledge, there is no dedicated method enabling systematic evaluation of the decision-supporting capabilities of RE tools. Our research contributes to filling this void by suggesting a summative, criteria-based evaluation method termed DESCRY\textsuperscript{4} (Decision-Supporting Capabilities of RE tools). The purpose of DESCRY is to investigate to what extent RE tools have decision-supporting capabilities. The method has a user-centred perspective, which means the evaluator should take the RE decision-makers’ perspective and estimate whether or not an RE decision-maker can perceive the existence of decision-supporting features as well as understand how to use it.

11.1 Methodological considerations

Since the purpose of DESCRY is to find out to what extent RE tools have decision-supporting capabilities, the evaluation is primarily summative. The results are, in the long term perspective, intended to have an effect on tools that support RE in general. Thus, our purpose is not that the evaluation should have an immediate effect on the qualities and features of the tools, which would have been the case with a formative evaluation. However, this would be a positive side effect.

We have also chosen DESCRY to be a summative merit evaluation. A summative merit evaluation is concerned with the intrinsic values of the evaluation object. Such an evaluation includes comparing the object with some form of standard (Guba & Lincoln, 1989). In the words of Cronholm and Goldkuhl (2003) who has concretized

\footnote{Apart from being the acronym for our evaluation method, it is also an English word, which means see a long way away or catch sight of.}
This for information technology (IT) systems, DESCRY is a criteria-based evaluation of an IT system as such. The main perspective of the evaluation depends on the criteria (Cronholm & Goldkuhl, 2003), which in this case is decision support for RE decision-makers. As a consequence, the evaluation model is user-centred. A user-centred evaluation applies criteria that are based on the expectations or the needs of the user (Vedung, 1998).

There are three appropriate data sources to use in a criteria-based evaluation of IT systems as such, namely a) the IT system, b) descriptions of the IT system, and c) descriptions of the criteria (Cronholm & Goldkuhl, 2003). In DESCRY, only the IT system and the criteria are prescribed to be used, and not descriptions of the IT system. The reason for this is that we have a user-centred evaluation in which the perceived affordance of the IT system is of main importance, whether or not the vendor claim that the IT system has certain affordances is of minor importance. The user has to perceive its existence and how to use it.

In the terms of Guba and Lincoln (1989), researchers of RE decision-making and decision support as well as RE tool developers are the agents of this evaluation method, while RE decision-makers and systems engineering companies are the beneficiaries. Concerning the agents, the evaluation method is aimed to be used by researchers with an interest in RE decision support and by developers of RE tools, with the purpose of providing better decision support to RE decision-makers. The evaluation method is not intended to suggest specific changes to an RE tool. Instead, it should be used in order to value the decision-supporting capabilities of RE tools and provide guidance of possible directions for future development and research efforts.

As described above, the evaluation method is criteria-based and the criteria become a bridge over the gap between what ‘is’ and what ‘might be’. The criteria in use are originated from desirable characteristics of REDSS (see section 9), which represent what ‘might be’ and the evaluation object is what ‘is’. Thereby, it is possible to identify the potentials of the tool.

Evaluations can be of different kinds of use. For example, an evaluation can have learning purposes or it can be used in decision situations. Thus, an evaluation can have controlling, promoting, or knowledge developing purpose (Vedung, 1998). The usefulness of carrying out evaluation using DESCRY is that it can promote better RE decision support by enhancing the body of knowledge of the decision-supporting capabilities of RE tools and thereby supporting learning concerning the potentials of RE tools to function as an REDSS.

11.2 Evaluation method - DESCRY

The design of the evaluation method consists of three steps:

1) Get to know the evaluation object
2) Systematically walk through the RE tool by answering the questions that are related to the criteria

3) Report findings

*First*, the evaluator needs to be acquainted with the evaluation object. The intended evaluation objects of this method are RE tools. Probably, many RE tools are not intuitive to use for a first-time user. This cannot be expected from such tools, since they are complex and are probably intended to be used frequently. Thus, evaluators should get to know the tool so that they can focus on the evaluation and not on manoeuvring it.

*Second*, the evaluator assesses the RE tool in relation to the criteria. The criteria are derived from desirable characteristics of an REDSS. For each criterion one, two, or three evaluation questions are formulated in order to facilitate the evaluation. The questions come from the guiding principles of the desirable characteristics. The walkthrough can be either task-based or criteria-based. In a task-based walkthrough, the evaluator chooses some basic tasks that he or she finds relevant for this particular evaluation. Then, the evaluator keeps the criteria in mind while conducting the tasks and thereby evaluating the RE tool. In a criteria-based walkthrough, the evaluator takes one criteria at a time and explores the RE tool in order to evaluate. The guiding star when to answering the questions is: *What is the likelihood that the RE decision-maker can perceive this affordance?* This means that the evaluator should not just identify if there are features that can fulfil a criterion. In addition, the evaluator should take the RE decision-maker’s perspective and estimate whether or not he or she can perceive the existence of the features and understand how to use them. Hidden and cumbersome features will most likely not improve the RE decision performance and will probably not satisfy the RE decision-makers. The criteria and evaluation questions are listed in section 11.3.

*Third*, the evaluator should put together and present the result in a reader-centred way, so that it is easily comprehended. For example, an overview of the result can be presented that shows which criteria that a) are fulfilled, b) to a large extent are fulfilled, c) to some extent are fulfilled, and d) are not fulfilled. The findings concerning each criterion should be described, i.e. the answers to the questions and their additional queries. The reader should obtain information about why a criterion is fulfilled or not and in what way the criterion (or lack of) is expressed in the RE tool. The evaluator is also suggested to formulate a conclusion by answering the main question of the evaluation, i.e., to what extent the RE tool has decision-supporting capabilities. It is also preferable to highlight the main strengths and main weaknesses of the RE tool from the perspective of RE decision-makers. By formulating such a conclusion, the reader of the evaluation result obtains a summarising statement of the decision-supporting capabilities of the evaluation object. The statement, together with an overview of the findings and a detailed presentation, will, potentially, make it easier to understand, value, and use the evaluation result.
11.3 Criteria and questions

For each question, we give some additional queries in order to provide examples of how the questions can be interpreted. However, it is not possible, and not desirable, to specify the exact interpretation of the questions since every evaluation situation is unique. Hence, the additional queries should not be viewed as the only way to interpret the questions. Instead, the evaluators should feel free to use the criteria and questions in a way that suit their purposes, the evaluation object, the RE decision-makers, and the tasks, i.e. the tasks that should be assisted by the tool.

C1) Reduce the cognitive load

Is it possible to obtain both overview and details?
• Can the RE decision-makers see the information details in a relevant context so that the understanding and use of the details are facilitated?

Is memory aid provided?
• Is it possible for the RE decision-maker to get or activate alerts; write and retrieve rationale for decisions; write and retrieve “soft” information (e.g. personal experiences, rumours, and opinions of others)?

C2) Ensure high usability

Are usability design principles followed?
• Are available functions and the status of the RE tool visible?
• Are design features used in a consistent way?
• Are the terminology and symbols familiar to the RE decision-makers?
• Can the RE decision-maker perceive how to use the functions?
• Can the RE decision-maker easily navigate in the system?
• Are there clear and logical mappings between controls and effects?
• Is feedback constantly and consistently provided?
• Are slips and mistakes rapidly and effectively recovered?
• Are there constraints that prevent inappropriate actions?
• Can the RE tool be used in a flexible way and is it possible for the RE decision-maker to personalise it?
• Is the design elegant and appealing?
• Is the interaction between the RE tool and the RE decision-maker pleasant, polite and friendly?

C3) Support availability of different types of information

Is the information mentally available, in terms of being visualised and easy to understand?
• Is the information visualised appropriately in relation to how it should be used?
• Is it possible to access information from outside the immediate environment of
the RE tool?
• Are there visual knowledge tools for pattern detection and knowledge
crystallisation?
• Is it possible to visually enhance objects?

Is information in different formats available?
• Is it possible to manage data in a database?
• Is it possible to gather, retrieve, classify, and manage unstructured
documents?

C4) Support different types of decision matters

Are decisions concerning requirements as such supported?
• Are system-related requirements decisions, e.g. requirements prioritisation, facilitated?

Are decisions of determining suitable ways to carry out the RE process supported?
• Are work-related RE decisions, e.g. choosing requirements acquisitions method, facilitated?

Are decisions that are made in other parts of systems engineering that use requirements as input supported?
• Are requirements-related decisions, i.e., beyond requirements decisions and RE decisions, e.g., test case selection, facilitated?

C5) Support creativity and idea generation

Are techniques that enhance creativity available?
• Is it possible for the RE decision-maker to be exposed to creativity enhancing information?
• Are brainstorming activities supported?
• Is idea generation in groups supported?

C6) Support knowledge sharing and transfer

Are there ways to share and transfer knowledge?
• Is knowledge of the application domain shared and transferred?
• Is knowledge of RE practice shared and transferred?
• Is knowledge of the developed system/system to be shared and transferred?

C7) Support idea evaluation and problem solving

Are evaluation techniques available?
• Is it possible to quantitatively compare the effectiveness of suggested alternatives?
• Is it possible to find out what will happen to a suggested solution if some aspect changes, e.g., an input variable, an assumption, or a parameter value?
• Is it possible to calculate which values of the input are required in order to accomplish a preferred level of a goal?

Is it possible to externalise a problem representation?
• Is it possible for the RE decision-maker to draw and make use of concept maps of problems, i.e., graphical representations in which concepts are linked to other concepts?

C8) Support decision communication

Are additional communication paths provided?
• Is it possible to communicate with decision stakeholders via the RE tool, e.g., via chat systems, interactive whiteboards, bulletin boards, shared information spaces, or virtual meeting systems?
• Is it possible to disseminate decisions to stakeholders?

Are negotiation facilities provided?
• Does the RE tool support bargaining, consensus seeking, or conflict resolution?

C9) Support coordination

Are coordination technologies available?
• Is it possible to manage interdependencies between activities to harmonise them?
• Is it possible to specify behaviours of the human actors, e.g., by establishing shared goals?
• Is it possible to plan behaviours of the human actors, e.g., by agreeing the set and order of tasks?
• Is it possible to schedule behaviours of the human actors by, e.g., assigning tasks to individuals or groups?

11.4 The decision-supporting capabilities of CaliberRM – Applying DESCRY

For illustration, we have applied the evaluation method on a commercially available RE tool. We chose to use Borland CaliberRM\(^5\) as object of evaluation for two reasons. One reason was practical; CaliberRM was easily available for evaluation purposes. However, the most important reason is that the vendor claims it to have “powerful decision support capabilities”, which made it interesting for the application of the

---

evaluation method. However, the choice of RE tool is of minor importance, since the evaluation method is applicable to any RE tool. The vendor’s own description of the tool is cited below:

“Designed for ease of use, the intuitive interface and powerful decision support capabilities of CaliberRM help teams deliver on key project milestones with greater accuracy and predictability. CaliberRM also helps applications meet end-user needs by allowing all project stakeholders — marketing teams, analysts, developers, testers, and managers — to collaborate and communicate the voice of the customer throughout the software delivery lifecycle.”

After getting acquainted with CaliberRM, we conducted the evaluation by making a criteria-based walkthrough. Hence, we started with the first criterion and answered its questions and additional queries by exploring the RE tool, then we went on with the second criterion, and so on until all criteria have been considered. After the walkthrough, the findings were put together. The results are presented in Table 4.

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Not fulfilled</th>
<th>Fulfilled to some extent</th>
<th>Fulfilled to large extent</th>
<th>Fulfilled</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reduce the cognitive load</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>The requirements information details are put into its hierarchical requirements context. There are also, e.g., traceability diagrams for overview purposes. However, there are no other types of overview and detail facilities available. Additionally, there is limited memory aid. It is not possible to activate reminding alerts, manage decision rationale, or manage “soft” information.</td>
</tr>
<tr>
<td>Ensure high usability</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>Usability design principles are followed to a large extent so that high usability is ensured. Available functions are visible. The design features are used in a consistent way. The terminology and symbols are familiar. It is possible to perceive how to use the functions. It is easy to navigate in the tool. There are clear and logical mappings between controls and effects. Feedback is constantly and consistently provided. It seems like slips and mistakes are rapidly and effectively recovered. There are constraints that prevent inappropriate actions. A drawback is that CaliberRM are not possible to personalise. The design is acceptably elegant and</td>
</tr>
</tbody>
</table>

---

6 In CaliberRM 9.0 Help → Getting started → Concepts → CaliberRM Overview
appealing. The interaction is pleasant.

| Support availability of different types of information | X | There are some visualisation possibilities, e.g., traceability diagrams. The information cannot interactively be visualised in an advanced way. It is possible to import Word documents, but there are no other ways to access information from outside the immediate environment of the RE tool. There are no visual knowledge tools for pattern detection. Beyond that pictures can be stored; it is not possible to visually enhance objects. It is possible to manage data in a database. It is not possible to manage documents. Although, it is possible to relate files to requirements. |
| Support different types of decision matters | X | System-related requirements decisions are facilitated to some extent, e.g., with help of traceability matrices and comparisons between requirements versions. However, there are no advanced techniques available to support requirements decisions. Work-related RE decisions are not supported. Requirements-related decisions are supported to some extent, since it is possible to, e.g., generate project estimates from the requirements. |
| Support creativity and idea generation | X | The RE decision-maker are not exposed to creativity enhancing information. Brainstorming activities are not facilitated. Idea generation in groups is not supported. |
| Support knowledge sharing and transfer | X | Knowledge of neither the application domain, RE practice, nor the developed system is shared and transferred. |
| Support idea evaluation and problem solving | X | It is not possible to quantitatively compare the effectiveness of suggested alternatives. Beyond traceability, it is not possible to, in an advanced way, find out what will happen to a suggested solution if some aspect changes. Project estimates are available. However, there are no other techniques available that makes it possible to calculate which values of the input that is required in order to accomplish a preferred level of a goal. It is not possible to draw and make use of concept maps of problems. |
| Support decision communication | X | It is possible for the stakeholders to discuss requirements. Contributions to a discussion are posted to requirements and the related stakeholders are alerted. Requirements change notifications are sent to stakeholders via e-mail. Negotiation facilities are not provided. |
| Support coordination | X | It is to some extent possible to manage interdependencies between activities. It is to some extent possible to specify behaviours of human
CaliberRM has some decision-supporting capabilities. The main advantages are that the social context of the RE decision-maker to a large extent has been taken into account. Decision communication as well as coordination is supported. Another advantage is that high usability is ensured, which is important in order to make effective and efficient use of the provided decision support facilities. CaliberRM has taken a step towards “powerful decision support capabilities” (as expressed by the vendor) in several other aspects. However, in our view there are several more steps to be taken before they can live up to that statement. We find CaliberRM appealing and there are potentials in it to develop into a tool that provide substantial RE decision support.

11.5 Chapter summary and reflections

In this chapter, we introduce a summative, criteria-based evaluation method, DESCRY, in order to illustrate a way the REDSS criteria can be used. The purpose of DESCRY is to find out to what extent RE tools have decision-supporting capabilities. The criteria and related questions are empirically as well as theoretically grounded. To obtain an indication of the usefulness of the evaluation method, we applied it to an existing RE tool; CaliberRM.

The application of DESCRY to an RE tool showed that the method has potentials to provide information concerning its decision-supporting capabilities. However, it is not possible to state the usefulness of the method by applying it to only one RE tool. Another limitation is that the evaluator and the inventor of the method (and the underpinnings of the criteria) is the same person, which can cause bias. To strengthen the knowledge of the potential usefulness of the method, we can compare it to evaluation methods of similar types in order to make a preliminary assessment of its strengths and weaknesses. In the research area of human-computer interaction (HCI), there are usability inspection methods, which are conducted by usability specialists in order to examine usability-related aspects of a user interface. The inspections are based on the judgments of the evaluators and the methods provide support in the form of, e.g., steps or heuristics (Mack & Nielsen, 1994). The resemblances between DESCRY and usability inspections methods are that they a) are carried our in the same manner, b) have a user-centred perspective, c) has the purpose of identifying characteristics of a tool, and d) is conducted by an evaluator without testing on real users. The differences are that a) DESCRY has a summative purpose and usability inspection methods have formative purposes, and b) DESCRY is specialised since it focuses on a certain type of tool and usability inspection methods are general in the sense that they are supposed to be useful for wide range of user interfaces.

A well-known usability inspection method in HCI is heuristic evaluation. In a heuristic evaluation, evaluators inspect the user interface. They go through the
interface, inspect different design elements, and compare them to a list of usability heuristics. They identify usability problems and make severity ratings. Then, the evaluators together summarise the findings and recommend actions (Nielsen, 1994). The advantages of heuristic evaluation are: a) it is quick, easy, and cheap, b) it identifies minor as well as major problems, and c) it can be used relatively early in the development process. The drawbacks are: a) it does not find all real usability problems, especially domain-specific problems are missed, b) usability specialists are needed in order to get major benefits, and c) it is subjective since heuristics are interpreted and the identified problems and severity ratings are based on judgments (Nielsen, 1994). The characteristics of heuristic evaluation imply strengths and weaknesses of DESCRY. Similar to heuristic evaluation, it is a method that is easy to learn and the evaluation is rapidly carried out (some hours). Thus, it is cheap to use. Concerning the use of the method early in the development process, DESCRY is not intended to identify problems in an iterative design process. However, it is possible to evaluate the decision-supportive capabilities of RE tools early in the development process of an RE tool. DESCRY can be used as early as in the conceptual phase, i.e., when the conceptual design of the RE tool are created and decided upon. Compared to heuristic evaluation, DESCRY has an advantage in that it is domain-specific. It is tailored for a certain domain, which makes it more precise. Just like heuristic evaluation, DESCRY needs specialists in order to get major benefits. Preferably, the evaluators should have knowledge of DSS and requirement engineering. At least, the evaluator needs to learn the basics of these areas. DESCRY is also subjective in the sense that it is analytical and not empirical. The consequence of not involving users is that we cannot be sure that real users should perceive the same affordances of the tool as the evaluators do.

The usefulness of DESCRY is not yet validated. So far, it has not been used by others than its inventors and its actual usefulness cannot be concluded without an assessment of the method with its intended users. This clearly requires further research. In addition, DESCRY is, as mentioned, summative, which means that it is concerned with the intrinsic values of the evaluation object, i.e. the RE tools. Hence, its purpose is not to suggest changes of the tools. Future research can be directed to transform the current summative evaluation method into a formative method. Such a method would make easier for an RE tool developer with the intention of increasing the decision-supporting capabilities of the tool to obtain concrete ideas for improvements.

To conclude the thesis, the research area of requirements engineering decision-making and decision support has taken some steps forward. Nevertheless, several more steps are needed before we can to claim to have a comprehensive understanding of the area.

Hopefully, our research, in the long term perspective, will contribute to the substantial improvement of the quality of RE decisions, greater RE decision-making efficiency, and requirements engineers experiencing an improved decision situation.
References


Anthony, R.N. (1965) Planning and control systems: A framework for analysis. Harvard University, Boston: Division of Research Graduate School of Business Administration


215


225


Silver, M.S. (1991) *Systems that support decision makers: Description and analysis*. Chichester, England: John Wiley & Sons


Appendix

Quotations in Swedish

1. … handlar ju egentligen om att man får ha någon form av indata som ska struktureras på något sätt och ska skrivas ner på den här lilla burken då, som man ska göra.

2. Ja, det är väl det man gör som kravansvarig: får in krav, skriver om dem, analyserar vad det är som gäller för oss. Diskuterar då med användaren och kollar upp, så att säga, att man har fattat rätt…

3. en kravansvarig givetvis ska vara duktag på sitt delsystem, men han måste också ha förståelse för hela vårat system för att veta vad för delsystem som finns runt omkring. [...] Och vet man då hur vårat system ser ut och vad det ska användas till kommer man att fatta snabbare och rättare beslut. [...] förutom att göra sin egen kravspec och ha koll på den så måste man förstå systemets kravspec för det är därför man drar ut kraven till sig delsystem tillsammans med den kravallokering som vi gör då också.

4. Öhh, ja, det är en massa utredningar i princip. Vi hittar något tvetydigt i systemkraven. Då måste vi prata med kunden, för att ta reda på vad det är de menar egentligen. Jag tycker att större delen av mitt arbete går ut på att göra utredningar.


7. … utreder funktioner. Vad det är som ska finnas där. Ofta så är det ju inte, tyvärr då, så är det inte bara vad utan det kommer in rätt mycket hur där.
Det går inte att komma ifrån, att man hamnar i den här hur-diskussionen samtidigt.

8. Men där är det ju också väldigt viktigt att man är... att man har en gemensam bild av vad som är svårigheterna i projektet. Så att man verkligen kommer fram till vad är det som är viktigt med detta.

9. En del löser problem genom att diskutera med andra och skaffar sig ett kontaktnät och löser det den vägen. Andra kanske sätter sig lite grann på sin kammare och går igenom... kanske läser den här överliggande kravspecen från pärm till pärm...

10. Ibland är ju hela utredningen en diskussion mellan honom och mig till exempel.


12. Utan istället får alla de här kravansvariga, kan vi väl säga, den här systemspecen och så är det upp till oss att prata ihop oss och så hår va. Så egentligen handlar det om att utgående från kravspecen som kommer uppför extrahera vilka krav det är som gäller på våran burk. Ensa ihop det med den historik vi har i form av att vi har andra konstruktioner som vi är likartade, så att säga va, och försöka återanvända dom och lite sådant där.

13. Man skriver ju ganska mycket hur produkten kommer att se ut. [...] så får man i ganska stor utsträckning själv konstruera på sitt system på hög abstraktionsnivå, som jag ser det. [...] Naturligtvis har vi hjälp av andra då och har mycket utredningar, men det är ju ändå den som skriver kraven som sätter samman alltihop till ett system.


16. Men jag tycker att det gäller och ha liksom en kommunikation hela tiden med de olika berörda instanserna då, vad som är viktigt och vad som är mindre viktigt. Försöka se till att man får det som kunden vill ha och...


19. Men som kravansvarig så har man informellt ansvar uppåt och få med alla krav ner, men man har ansvar neråt att se till att alla krav kommer in i kravspecen som konstruktörerna, eller ja, han som designar delsystemet då. Man har formellt ansvar neråt, men informellt ansvar uppåt.


22. Som kravansvarig så jobbar man i… man sitter med sådana här riskanalyser, kallar vi det för. Då går man igenom… man bara sätter sig alla och så brainstormar man om vilka risker man ser i projektet.

23. Det här är det största sättet att förmedla information mellan delsystemen, att man blir kallad till granskning.


Sen är det ju också vilken nivå man ska ha kraven på. Det kräver väldigt mycket erfarenhet också. När man pratar om användningsfall så brukar man prata om att de kan ligga på bottom level och surface level och uppe i himlen.


Ja, vilka krav som är väsentliga överhuvudtaget för huvudfunktionerna, vilka är... Vilka krav är väsentliga nu? Ha en viss förståelse för prioriteringsordning, när gör man vad. Och även är det ju prioritering av hur djupt ska jag gå. För man kan inte gå lika djupt ner och fundera på... för varje krav. Man måste få en känsla för, var ska jag lägga min tid.


Det första, den viktigaste bedömningen, den gör vi redan från start alltså. Det är ju att sätta någon gräns, för hur långt man ska utreda någonting. Vilken ambitionsnivå man ska ha och det är ju en bedömning som man bara... Den blir lättare och lättare med erfarenhet. [...] Utan du får ju liksom... tillräckligt bra, den bedömningen tycker jag personligen är jättevår.

Det kan typiskt en kravansvarig göra då och det kan man dokumentera i form av rapporter, korta rapporter eller långa rapporter, mail eller dragningar eller ibland i krav. Men det är inte alltid det resulterar i krav.

Då satte jag mig ner och gjorde små rapporter på hur funktionen skulle fungera för de olika funktionaliteterna. [...] Vilka avvägningar har vi gjort?

33. Sen så är man med och granskar. Man kallas till alla granskningar av designspecar gör man även som kravansvarig.

34. En sak som man får göra en hel del är att hjälpa verifierarna och tolka kraven så att det blir rätt. Det är väl samma med konstruktörerna, som behöver en del tolkningshjälp. Man skriver inte alltid världens bästa krav som är entydiga och lättförståeliga.

35. … både kravarbetet, kravingenjörerna och kravansvariga är ju en servicefunktion på något vis i projektet kan jag tycka. Hela tiden... man ska hålla koll på vad som händer runt omkring och man ska svara på frågor och...


37. Och... men jag läser ju deras testspecar, så att säga. Jag är ju med och granskar deras testspecar, hur de tänker testa kraven.

38. När det gäller MMI:t så validerar man ju även då med kunden, eller ja, inte med kunden, utan med användaren faktiskt.


40. Man kan upptäcka det själv som kravansvarig eller man kan få det från projektledning eller systemledning att ni är på väg åt fel håll eller har ni verkligen koll på det här. Så att vem som initierar eller vad som initierar det kan vara... ja, det kan vara akuta problem systemet eller konstruktionsarbetet, eller planerat.

Nej, i en utredning så kan det ingå allt möjligt som kalkylark i Excel och till avancerade matematiska modeller. Man försöker att modellera så gott det går hur det här kommer att bli i verkligheten när det väl är infört, så att det gör vi mycket sådana väldigt avancerade modeller och simuleringsverktyg.

Kommer det här att påverka liksom hur systemet beter sig utåt? Hur kommer det här att påverka vad kunden upplever? Det är väl i regel den första frågan man ställer sig. Behöver vi överhuvudtaget ha det här kravet eller kan konstruktörerna få göra som de vill?

Ja, det är ju att det finns en bakgrund och att den är heltäckande och det finns en bakgrund och orsaken till att man vill göra som gör. Att det är förtydligat och förklarat vad de vill... man göra. Det som man kanske kan tycka är svårest att göra, det är ju i slutänden en kostnadsskattning. Vad kostar det att införa det hela?

Ja, det beror på vad det är för förslag. Ofta kommer det ju ett sådant förslag med en intern utredning, att man har bestämt i förväg att göra en utredning och sen tar man en granskning i så fall. [...] Skälen till att göra det, lönsamheten, vilka som blir berörda, när det ska införas och...

De allra flesta problemrapporterna på krav kommer från verifieringen när de sitter och gör sina verifieringsspecar och när de testar så att säga. Sen var egentligen då ursprunget är... Många gånger hamnar man i diskussioner med att man måste diskutera med kund. Att man har hittat liksom ett vitt hörn i systemet eller vad man ska säga, som egentligen inte är specat hur det ska bete sig.

Det är ju rätt intressant att när det blir fel, när det inte funkar i verifieringen, när det händer något däremot där. Då blir det i regel... ja, då felrapporterar man gärna kravet.

Ofta när vi får felrapporter så har vi en massa indata från kund som man måste gå igenom.

man får in något problem och sedan i någon form av forum antingen i med *mail* eller att man samlas och diskuterar... försöker man hitta hur det systemet ska bete sig och hur man löser det enklast att komma dit. Det är
väl egentligen så man måste jobba och det är väl så vi jobbar, utan det är 
mer då att det ibland tar tid och att folk inte har tid och så där kanske mer 

50. Nej, det kan ju till exempel vara ett krav som har uteblivit. Det borde 
ställas krav på någotning och det noteras ju nästan som om det är ett 
felaktigt krav. Att man gör ett förslag på ett nytt krav och sen blir det 
granskningstillfälle så tar man upp det till bedömning och för in eller 
ändrar det eller vad man nu gör. Man bestämmer hur det ska hanteras i 
alla fall. 

51. Jag menar, systemet måste ju hänga ihop, så att i den mån det slår på andra 
krav så får man ju försöka ta med det också, så att säga. Vi sitter ju inte och 
bockar av det liksom mot alla krav, men... utan det är väl mer upp till våra 
kunskaper i huvudet, så att säga. Att man känner att, okej, men då blir inte 
det här bra och hur hänger det ihop med det där och... 

52. Tack vare att vi har det här förnämliga [kravhanteringsverktyget] och 
databassystemet och så där, som jag sa tidigare, så märker man att när man 
ska gör någon liten käck, trevlig ändring bara som är så här... Det här är 
alltså en kravatom som sitter i sju stycken olika projekt och på något sätt så 
ska det då förankras i alla dom har sju projekten och alla dom har sju 
projekten ska diskutera huruvida dom ska föra in den här ändringen eller 
om dom inte ska föra in den här ändringen, om dom har råd att föra in den 
här ändringen, i vilket skede dom är... 

53. om det är någon rätt stor rättning så måste man bestämma sig när ska vi 
föra in den här rättningen, i vilket utvecklingssteg? Ska vi gå in med en 
gång och ändra eller ska vi ta in det senare? Det här är ett rätt stort beslut 
man får fatta. 

54. Det kan vara en olycklig språkgroda som gör att kravet är svårt att förstå, 
ja, då uppdaterar man ju bara och rättar den. Å andra sidan så kan det vara 
någotning som innebär en större omkonstruktion. Då får vi lyfta oss till 
lämplig nivå i innan det beslutas. 

55. Ja, då åker det in i specen som revideras och sen fördelas det ut till 
delsystemen som är berörda utav det. 

56. Ja, då blir notum. Det noteras bara, antingen blir det granskningsprotokoll 
eler redan i felrapporten kanske. 

57. Sen genererar jag ut lite deltaskrifter också till mina verifierare och 
konstruktörer som precis talar om vad som hänt mellan olika revisioner 
samt där. 

58. Ja, det är olika processer. Det tycker jag. När du har fryst kraven... innan 
du har fryst kraven, då håller du på med etableringen. Men när du har 
fryst kraven kommer du in i kravändringen.
59. Det stämmer väldigt bra med hur det ser ut i dagsläget, som ett *snapshot* av vår verklighet.

60. [Informant A:] Ja, vid uppstart av ett projekt rent generellt så finns det väl två vinklingar på det där. Det ena är väl att det är... man har någon slags *legacy* från föregående projekt av samma typ. Att projektet tar fram en ny modell av någonting eller något sådant där och då ärver man ju krav och sedan finns det den delen att man skapar nya krav som gäller för just den här nya grejen, så att det... Det är två grejer jag inte... Etablera krav... Jag vet inte om man ska dela upp det eller om det innefattas i en punkt där. [Informant B svarar:] Jag tycker nog då att även fast det är krav man har, så blir det ändå att man samlar ihop dem kraven och får dem att gälla, oavsett var de kommer ifrån, så finns det alltid några grundkrav och sen så är det några nya krav och sen är det några vi inte vet. Och sedan paketerar man. Det är lite grann paketering hela den modellen.


62. Jag tänkte just för *searching existing components reuse* känns ju som att det är en viktig del man gör just under kravförändring, andra spalten där. [...] Jag ser inte det lika tydligt, så att säga, som det står här i *establishment of requirements*, men... [...] Jag skulle vilja byta dem. ... eller med min erfarenhet, så känns det mer som att man jobbar på det viset här än att man... *search and screen* är mer en bit in projektet så att säga under någon mer... Inte under en tidig fas utan liksom... Om det är så som jag uppfattar som du förklarar *search and screen*.

63. Jag har en känsla av också att man befinner sig många gånger också i gråzonen mellan de här [...] så det blir en mix av den här tidiga *establishment* och *management*.

64. Jag tror nog att det ställer... högst oftast är så för att om man redan har någonting som kommer in och det kommer in ett ändringsärende på den, så är det oftast inte så stora grejer så att det är värt att kasta bort det man har gjort och leta efter någonting nytt annat. Så ur den synvinkeln stämmer det nog rätt bra.

65. I mitt fall stämmer det bra. Den här tyngdpunkten ligger nog i den här etableringen...

66. Det handlar väl rätt mycket också vad det är för slags resultat som är *output* av projektet. [Företaget i fallstudien] förstår jag är väldigt noga med risikanalys och så där, även om det kanske inte syns då på kravändringen.
[...] Medicinteknik vet jag också är väldigt noggranna även vad det gäller hantering av ändringar att det görs analyser och spåras och allt möjligt. [...] medan andra branscher inte alls håller på med sådant, utan bara ändrar. Det är nog väldigt branschberoende. ... hur produkten ser ut i slutändan.

67. Vad gäller ändringar så är väl oftast en ekonomisk analys man genomför. Är det här värt att göra?

68. Ja, det finns för- och nackdelar med det naturligtvis, men jag tror att det är så.


71. Den känns som att det är under den här fasen som man spottar ur sig en massa felrapporter och andra typer av återmatningar så att säga. Dokument i det här fallet kan ju vara testrapporter och felrapporter och prestandavärden och sådant [...] återmatning, men som knyter an till kraven.

72. Det finns folk på den enheten där jag jobbar som på allvar säger att det där är bara blaj, det ska vi inte syssla med.

73. Det verkar inte vara så där väldigt populärt för konstruktorer att övergå till antingen krav eller verifieringen. Det egentligen borde krävas att man har kanske jobbat ett tag och fått lite erfarenhet då innan man kan... För det är ju svårt, liksom att man behöver ju ha ett nätverk inom företaget för att kunna förstå vilka personer det är man ska prata med i ett projekt när det gäller att samordna olika saker. Som nyutexaminerade så känner du ju inga i allmänhet då som jobbar på företaget.

74. ... det har inte varit alltid så attraktivt för konstruktörerna att gå till det jobbet. [...] Och då har man rekryterat utifrån direkt till den här rollen då.

75. Många av de här som har kommit in som kravansvariga har kommit till [företaget] och inte har jobbat som tekniker innan och inte jobbat i organisationen innan, och har då kanske inte den här förmågan att... ha tillräckligt med auktoritet i sin roll och den auktoriteten får man ju också bara genom att visa att man kan det man håller på med, alltså genom kunskap. Sen så tror jag att många av de här (dvs. kravningenjörerna) har alltid varit ett halvt steg efter och då är det jättesvårt att komma ifatt och ligga ett halvt steg före istället.
76. [Informant A:] Men att gå på kurs för att lära sig om krav, det finns inte idag. [Informant B:] Nej, eller hur gör man i de här komplexa situationerna då liksom, då när man begär vissa saker, kravställa vissa saker. [Informant A:] Det väl att man har fått för sig också att kravhantering är inget att satsa på. [Informant B:] Det är ingen kompetens.

77. Och tala inte om för mycket, utan tala om bara precis det vi behöver veta. Men tala inte om mer åt oss, än precis det vi vill veta. [...] alla vill ju ha den här burken som är gränsen och man vill ha hjälp med vissa saker, men vissa saker vill man inte att någon annan ska lägga sig i.

78. [Informant A:] Finns det mycket prestige mellan grupper? [Informant B:] Jo, men om man ändrar i organisationen blir det ju att varje sådan ny organisation ansvarar för sin del. Ingen i den nya organisationen vill bara vara resurspool, utan man vill ha ett ansvar för ett delsystem.


80. Det är att vissa, vi pratar delsystem här, drar på sig fler problem på grund av att andra delsystem mer ogärna vill förändra sig. Även om något inte skulle passa bättre i ett annat delsystem, så blir det det här medgörliga delsystemet som tar på sig ansvaret att göra det här.

81. … de där reviren igen, man får ju liksom inte prata om saker inuti det där delsystemet till exempel. Då känner de sig ju påhoppade på något sätt av någon anledning.

82. Man kan inte vara med ända från kundfasen, men i alla fall i systemfasen, i analysen och allthopå hur man fördelar, varför gör man det, varför ser systemet ut som det gör. Det borde ju finnas folk ifrån alla andra delsystem med och inte bara få det serverat sen. – Varsågod. – Vad är detta? – Mer behöver du inte veta.

83. Jag kan nog gå in i princip på vilka use case som helst och se vilken person det är som har varit och skrivit i det. Så att det är på liksom den... Vi har så olika sätt att skriva det på det alltså.

84. Men vårt problem är lite grann att vi kan ha placerat oss på detaljnivå i vissa fall och i andra fall är det för abstrakta krav. Det finns inget ensamt sätt att skriva egentligen.

86. En brist här är då att vi kravansvariga i delsystemen inte var med tidigare i kontakterna med kund, när kundens krav specades. För att för det först styra upp det hela så att vi fick lite klarare krav och för det andra förstå vad kraven egentligen menades och vad kraven avsågs. [...] Antaganden som inte vi känner till, som gör att våran tolkning inte är vad kunden håller med om.

87. För att få en överblick över vad som ska implementeras vore det väldigt bra att få vara med från början.


89. Ha, ja egentligen frågar du hur lång tid det tar att processa våra problemrapporter och är det sådana där saker som inte liksom är att nu funkar det inte, utan det är mer hur ska det funka, jag menar det kan ju ta ett halvår innan det liksom alla har fått säga sitt och man har ångrat sig tio gånger och det kan ju ta väl långt lång tid med våra problemrapporter innan de är avstädade. Inte för den sakens skull att det liksom är tusen timmars effektivt arbete, utan det är väldigt få timmar effektivt arbete, det är det va. Men det drar ut på tiden och det är massa människor som tycker en massa i början och sen efter ett tag kommer någon projektledare på att, nja, det här har vi inte pengar till. Det vill vi inte ändra. Det kan ta lång tid.

90. Ja, det är de här långbänkarna som aldrig kommer någonstans, så att säga, så man får aldrig ner foten riktigt och när det liksom... [...] Det tar lång tid och system pratar väl med kund och det kan vara långbänkar som är väldigt sega och under tiden konstruerar vi.

91. Vi fattar beslut, vi dokumenterar dem inte och vi sprider dem till fel människor. Där kan vi bli mycket bättre.

92. Där har vi ett jättestort problem att vi inte har dokumentation på varför det blev som det blev eller varför vi gjorde som vi gjorde.

93. ... vissa detaljer åker runt i organisationen och är jätteviktiga, men vi glömmer stora tekniska problem som ligger som tickande bomber och som vi borde tagit tag i på en gång.

94. Prestendakrav är ju egentligen ganska kritiska, men de känns inte så, tyvärr. [...] Vi har inte tänkt på prestandaoptimering när vi gjorde vår
konstruktion. Det blev bara... Vi har bara prioriterat funktionaliteten. Det kritiska är ju inte då att skriva det här, föra in det i [kravspecifikationen], utan verkligen att uppmärksamma folk på att det finns krav.

95. Sen, vi kan ju inte suga åt oss information, utan... Jag kände inte ens till att det här protokollet fanns innan det var dags att implementera. [...] Någon borde ha haft, borde ha känt det som sitt ansvar att förmedla information om att det här finns, det här kommer att komma in sen.

96. Vi har alltså en massa krav som ingen tänker på förrän vi ska verifiera sådana [...] Det hade varit lite mer kritiskt kanske. Informera folk om att det faktiskt finns krav.

97. ... just RTM det är en hög tröskel att börja jobba med det. Många gillar ju inte det då. Det är en träningssak alltså, dels att börja tänka... tänka i något strukturerat tankesätt. Det är man inte van vid heller. Så det är en stor tröskel att... med detta. Det är lite bekymmersamt. Men det är alltså så att verktyget är inte riktigt... Man är så van att det ska klickas och grejas och så här då va, och hela tiden få väldigt mycket återmatning var man är och vad man gör. Det ger inte det här verktyget, utan du har din lilla prompter och dina små rutor som du ska fylla i. Man har svårt att se helheten då kan man väl säga.

98. ... hur det här kravverktyget användes för tre, fyra år sedan då... och det ledde till dokument som vi som inte varit med tyckte var helt oläsbara.


100. För att ett intuitivt och lättanvänt verktyg som... Det tror jag kommer fortare att spridas i en verksamhet än om man har ett besvärligt, svårhanterligt verktyg [...] som avsiktligt lägger krokben för en så man är riktigt trött. [...] Men själva grunden med kravhantering i databasform, det tror jag är fullständigt självklart.

101. Ja, jag upplever inte den som väldigt intuitiv och så, även om det har naturligtvis många kraftfulla funktioner, så hade jag gärna sett intuitivt på en nivå så i Access.

102. Så det känner jag, jag går väldigt mycket på känsla, har jag fått göra. Kolla av med henne och om det är helt uppåt väggarna.

103. ... vad som saknas tycker jag är den där omhändertagande, inskolning i arbetsuppgifterna.

104. Det är ju alltid problemet när folk är nyutexaminerade. De har ju en uppläningstid. Man underskattar nog ofta den. Det tar ju... för att bli
väldigt bra i sin disciplin så krävs det helt enkelt flera års arbete så man har blivit lite erfaren. Så är det ju ganska stor omsättning på personal.

105. Deras grundproblem är nog att de alltid kommer att hamna i det här mellanläget de får underlag från system för sent och att konstruktionen måste igång för tidigt. Så att ofta så den tidsluckan som de skulle behöva ha för att göra sitt arbete är blir svår att...

106. … till slut så rinner tiden iväg och mållinjen när allt ska vara färdiglevererat närmar sig ju och då måste konstruktionen börja i förtid. […] på den här enheten att man är glada konstruktorer och jobbar på hörsägen […] Att man berättar hur det ska fungera och så bygger någon efter det.


108. Men det är ju alltid frustrerande för dem som skriver kraven och de som verifierar, för som konstruktor så tjänar man väldigt mycket på att kraven är fel. […] De ser det väl ofta inte som ett stöd i arbetet, utan som ett hinder för att… som ställer till det.

109. Är det mycket konstruktorer tycker de att det är för lite detaljer alltid och är det från system så kan de tycka kanske att det är för mycket detaljer.

110. Man ser en massa träd, men man ser inte skogen.


112. Det är lite dumt det här, för att kundkraven finns ju på svenska och så översätts de till engelska och det tycker jag personligen är olyckligt när vi är svensktalande allihop. […] Alltså, först översätta… någon översätter det och sen så måste jag översätta det tillbaka för att förstå det. Då har jag kanske inte tolkat det på samma sätt som vederbörande som skrev det.

113. [Informant A:] För det kan jag… när man sitter och jobbar med krav, det kan jag sakna att man inte ser beroendena emellan kraven. Det kan ju faktiskt vara så att en kravändring här, det låter inte så himla märkvärdigt att ändra den här grejen, men om man gör det så var det några andra saker där borta som klickade till och det har du inte koll på förrän du kommer så långt som till… Det kanske till och med gått förbi konstruktion. […] Och det kanske inte märks att det är fel förän man slår ihop det då va. [Informant B:] Nej, det känns ju som att det tar mycket plats i huvudet på en, alla kraven. [Informant A:] Man får ha mycket i huvudet och det går inte att få ner allt. [Informant B:] Det är väldigt svårt. Inte på annat sätt än
att tänka på hur kan det här påverka dom övriga som man ska minnas ganska väl.

114. Sen beslutsbiten, då är det ju det här som jag nämnde innan då att man måste ha alla beroenden i huvudet. Det är ett beslut som jag tycker är svårt när man ska fatta beslut om kravändringar…


119. Det som inte stämmer med min erfarenhet, det är väl det med nyutexade, grönningar, som sitter på det där, utan jag kan väl ofta se att det kan vara gamla rävar som sitter där, men…

120. Nej, inte nyutexade, det tror jag aldrig jag har stött på…

121. Men jag upplever nog inte att det är ett lågstatusjobb så att säga, utan jag skulle nog säga att det är hög status på dem som jobbar med… Det är ändå de som får sätta ner fötterna och bestämma sig för hur det ska göras. Och de är ofta i de flesta fall väldigt kunniga. Det är sällan som man har någon som är väldigt oerfaren och inte kan någonting så att säga.

123. … däremot så att krav är skrivna på olika sätt av olika personer, det har man ju drabbats av, vilket gör att det kan vara svårt att tolka eller att de helt enkelt inte går att verifiera, vilket ju är centralt för en testare.

124. … att en kravhanterare ses väl rätt ofta som en administrativ, icke-teknisk person eller roll och det är väl att de hamnar utanför när det ska pratas om tekniska beslut och sådana saker, så att jag tror att det stämmer väldigt bra det här med att de inte är inblandade [dvs. i diskussioner].

125. Ja, jag skulle nog säga att om de som skriver kraven skulle kommunicera ut det bättre så skulle det ju åtminstone landa samtidigt i implementationsfasen som i testfasen. [...] Jag tycker att kravställarna har en möjlighet att sprida information till så många som möjligt om när kraven är uppdaterade, så att det landar hos alla samtidigt.


128. Jag skulle vilja säga att så som jag upplever kravhantering så är det väldigt mycket av ett mind set, liksom att man ska tänka på ett visst sätt, som man kanske inte gör i en annan ruta i närliggande processer, utan man måste ha ett annat fokus.


130. Men tyvärr, så det här verktyget gör att dem hamnar huller om buller, så det blir väldigt svårt att läsa igenom och få ett grepp över det.


Ja, bästa möjliga observerbarhet. Sen är det ju frågan om hur lättanvänt verktyget är. Och där vill man ju alltid att det ska vara mer lättanvänt. Mera intuitivt.

Det tycker jag är ett starkt önskemål. Mera intuitivt handhavande.

För kravhantering i databasform eller motsvarande tycker jag nog är absolut nödvändigt och nästa steg är ju att det ska vara användbart, hanterbart och därmed tillgängligt för alla som berörs utav det. Det tycker jag är det klart viktigaste.


Med så många som slutar, så finns inte den kunskapen kvar.

… när man har mycket omsättning på folk och det kommer mycket nytt folk och så vidare. Och få hålla uppe kvaliteten på det är väldigt svårt alltså.

Det är väldigt lätt att tro att bara vi har ett verktyg så ordnar det sig. Men man måste ju veta vad man vill göra innan man skaffar ett verktyg, så att säga va. […] man ju titta på hur kan ett verktyg stödja oss i det arbete. […] För det var liksom bara en, jaha, här är en databas, gör något bra med den och det kunde inte folk, så att säga. […]Sen är det ju alltid faran med verktyg också att det är väldigt lätt att säga man matar in det här och det här också för varje. […] Man får passa sig så man inte gör ett administrativt monster, så att säga, som ska matas med siffror varje morgon. […] Men innan man gör det så måste man veta… orkar vi underhålla det här eller kommer det här bara att bli under entusiasmens fagra vecka, så att säga, som vi lägger in och gör kravspårning till exempel. Kommer vi att hålla det här uppdaterat eller blir det bara en gång och sen blir det inte mer.

Men inte stoppa in allt möjligt roligt [in i verktyget] för att det går.
<table>
<thead>
<tr>
<th>No</th>
<th>Author</th>
<th>Title</th>
<th>Year</th>
<th>ISBN</th>
<th>DOI</th>
</tr>
</thead>
<tbody>
<tr>
<td>192</td>
<td>Dimiter Drinkov</td>
<td>Towards a Many Valued Logic of Quantified Belief, 1988</td>
<td>1988</td>
<td>91-7870-374-3</td>
<td>[Online]</td>
</tr>
<tr>
<td>214</td>
<td>Tony Larsson</td>
<td>A Formal Hardware Description and Verification Method, 1989</td>
<td>1989</td>
<td>91-7870-517-7</td>
<td>[Online]</td>
</tr>
</tbody>
</table>


No 808  Klas Gäre: Tre perspektiv på förändringar och förändringar i samband med införande av informa-


<table>
<thead>
<tr>
<th>No</th>
<th>Author</th>
<th>Title</th>
<th>ISBN</th>
</tr>
</thead>
</table>

**Linköping Studies in Information Science**

<table>
<thead>
<tr>
<th>No</th>
<th>Author</th>
<th>Title</th>
<th>ISBN</th>
</tr>
</thead>
</table>

No 13 **Stefan Holgersson:** Yrke: POLIS - Yrkeskunskap, motivation, IT-system och andra förutsättningar för polisarbete, 2005, ISBN 91-85299-43-X.

No 14 **Benneth Christiansson, Marie-Therese Christiansson:** Mötet mellan process och komponent - mot ett ramverk för en verksamhetsnära kravspecifikation vid anskaffning av komponentbaserade informationssystem, 2006, ISBN 91-85643-22-X.