12-4-2011

Coping with Requirements Uncertainty: An Interpretive Case Study

Marcus Keutel

Bjoern Michalik

Werner Mellis

Coping with Requirements Uncertainty: An Interpretive Case Study

Marcus Keutel
University of Cologne
keutel@wiso.uni-koeln.de

Bjoern Michalik
University of Cologne
michalik@wiso.uni-koeln.de

Werner Mellis
University of Cologne
mellis@wiso.uni-koeln.de

ABSTRACT

Information systems development projects (ISDPs) often face requirements uncertainty. Thus, coping with this issue is an important project management task in ISDPs. In order to gain insights into which different requirements uncertainty situations exist and how to cope with them, we conducted a longitudinal interpretive case study. We investigated the requirements analysis of an ISDP in an international insurance company for 17 months. Thereby, we identified eight distinct requirements uncertainty situation types, observed ten requirements engineering (RE) techniques applied in practice and their impact on requirements uncertainty situation types, and finally derived recommendations for applying RE techniques in requirements uncertainty situations based on our observations. We combine these findings and contribute to RE literature by making a first step towards a framework for coping with requirements uncertainty in ISDPs. We believe that RE researchers as well as practitioners in ISDPs can make use of this article's insights.

Keywords

Requirements uncertainty, requirements risks, case study, information systems development, decision making.

INTRODUCTION

Requirements uncertainty, also called requirements risks, has been in the focus of information systems (IS) research for the last two decades (e.g., Barki, Rivard and Talbot, 2001; Beynon-Davies, Tudhope and Mackay, 1999; Bhattacharya, Krishnan and Mahajan, 1998; Hsu, Chan and Chen, 2008; Lyytinen, Mathiassen and Ropponen, 1996; Mahmood, 1987; Mathiassen, Tuunanen, Saarinen and Rossi, 2007; Mellis, Loebbecke and Baskerville, 2010; Moynihan, 2000b). Requirements uncertainty potentially leads to inadequate software solutions, rework, or delay and is closely related to project success (Burns and Dennis, 1985; Ebert and de Man, 2005; Han and Huang, 2007; Saarinen and Vepsalainen, 1993; Stephenson and McDermid, 2005; Wallace, Keil and Rai, 2004a). IS scholars propose a variety of single requirements engineering (RE) techniques (Cheng and Atlee, 2007; Mathiassen et al., 2007). However, all these techniques are stand-alone solutions and only cope with very specific requirements uncertainty aspects.

Contrarily, Mathiassen et al. (2007) propose an integrated framework to cope with requirements uncertainty on a project level. However, analyzing requirements uncertainty on this abstract level is very difficult to operationalize. Thus, we follow Mathiassen et al.'s call for empirical research and make a first step towards a framework which is still integrated but works on a situational level.

Our objective is to develop a framework describing how varying RE techniques affect different situations of requirements uncertainty. We develop this framework by answering the following research questions (RQs):

RQ1: *How can different situations of requirements uncertainty be characterized in practice?*

RQ2: *How do practitioners cope with these situations regarding the use of RE techniques?*

RQ3: *How successful are the applied techniques in coping with these situations?*

To answer these research questions, we apply a case study research (CSR) approach. CSR is an adequate methodology for our study for the following reasons. First, scholars recommend applying CSR in exploratory studies (Eisenhardt, 1989). Additionally and according to Yin (Yin, 2009), CSR is suitable to answer research questions of ‘how’. Finally, we follow the call for “case studies of the relationship between practices and techniques” (Mathiassen et al., 2007, p. 583) and the effects that techniques have on requirements uncertainty. CSR allows gaining rich, contextual insights into the dynamics of phenomena under investigation (Dyer and Wilkins, 1991), in our case the RE practice of coping with requirements uncertainty.

We contribute to existing RE literature by making a first step towards a framework to support practitioners in coping with requirements uncertainty situations. The framework consists of three parts: Requirements uncertainty situation types, RE techniques, and recommendations for applying RE techniques in requirements uncertainty situations.

The remainder of this paper is organized as follows. In the subsequent section, we briefly discuss the literature on coping with requirements uncertainty. Then, we explain the research methodology underlying our case study. Afterwards, we describe the framework derived from our case study, before we discuss our findings and present implications for practice and research.

THEORETICAL BACKGROUND

Requirements are descriptions of the future system’s functions, features, properties, or expected behavior (Sommerville, 2011). They are derived from the user needs. The concept of requirements uncertainty has been widely studied in the IS literature and is also well known in practice (Jones, 1996). Derived from the concept of uncertainty in organizational science (Galbraith, 1973), requirements uncertainty is defined as an information deficit originating by the difference between the amount of information needed for specifying the requirements and the amount available to the analysts (Na, Li, Simpson and Kim, 2004; Nidumolu, 1996).

Numerous works investigate ways to cope with requirements uncertainty in software development projects (e. g., Davis, 1982; Fazlollahi and Tanniru, 1991; Hsu et al., 2008; Mathiassen and Pederson, 2008; Mathiassen and Stage, 1990; Moynihan, 2000a, 2000b; Nidumolu, 1995, 1996). Thereby, many articles propose single specific techniques to cope with requirements uncertainty, for instance creating mutual understanding (Champion, Stowell and O'Callaghan, 2005; Vlaar, van Fenema and Tiwari, 2008), documentation (Moynihan, 2000b), fostering social interaction (Hanisch and Corbitt, 2007; Harris, Collins and Hevner, 2009), prioritization (Port and Bui, 2009), prototyping, including mock-ups, simulations, screen painting and dummy printouts (Benaroch, Lichtenstein and Robinson, 2006; Galal, 2001; Moynihan, 2000a), requirements workshops (Hickey and Davis, 2004), special interview techniques (Pitts and Browne, 2007), and video conferencing (Hanisch and Corbitt, 2007). However, all these techniques are stand-alone solutions and only cope with very specific requirements uncertainty aspects. The overall requirements uncertainty situation is not examined.

Contrarily, Mathiassen et al. (2007) propose an integrated framework to cope with requirements uncertainty on a project level. They identify three different requirements development risk dimensions and four types of requirements development techniques. By assessing each of those three risk dimensions on a high or low scale, they build requirements development risk profiles. For each of those requirements development risk profiles, they finally propose one or several of the requirements development technique types. Thus, the application of this framework is an iterative process throughout the whole IS development project (ISDP). For every change of a requirements development risk profile, the applied requirements development techniques have to be modified.

Although this is a well-elaborated, integrated, and thus revolutionary approach, we think that assessing the requirements development risk profiles and applying the requirements development techniques on a project level throughout the whole ISDP is very difficult to operationalize. Thus, we follow Mathiassen et al.’s call for empirical research (Mathiassen et al., 2007). We continue their work by empirically analyzing requirements uncertainty on a detailed situational level. Thereby, we investigate situations where requirements uncertainty concerning single requirements emerges. We aim to develop a framework describing how varying RE techniques affect different situations of requirements uncertainty. We consider these insights to be useful for further RE research and practice.

METHODOLOGY

In this section, we introduce our case study's research design and the research site.

Research Design

We describe our research design with regard to the following criteria: (1) philosophical foundations, (2) theorizing, (3) case selection, (4) data collection, (5) data analysis, and (6) researchers' involvement. We discuss more details about the methodology-in-use of our case study in another paper (Keutel and Mellis, 2011).

Philosophical Foundations

We position our work as an interpretive case study. Interpretive research relies on the assumption that people create and associate their own subjective and intersubjective meanings as they interact with the surrounding world (Dyer and Wilkins, 1991; Orlikowski and Baroudi, 1991; Walsham, 1995, 2006). Consequently, interpretive researchers understand the world under investigation and themselves as not separable. Thus, they attempt to understand phenomena by accessing the meanings that participants assign to these. The researchers are aware that the data gathered are their own constructions of other people's constructions of their perceptions of the world.

Theorizing

Following Ragin (Ragin, 1997), we use case-oriented theorizing. The value of the case-oriented approach is its ability to produce holistic and particularized causal explanations for the outcomes of each investigated case (Piekkari, Welch and Paavilainen, 2009). In this case, theorizing means "tracing the causal processes that generate outcomes in specific contexts" (Piekkari et al., 2009, p. 571). Especially, the context of a phenomenon under investigation is thus regarded to be very important to derive meaningful explanations. The generalization takes place within a single setting instead of generalizing a theory across different settings (Geertz, 1973; Lee and Baskerville, 2003).

Case Selection

Single case studies allow rich, contextual insights into the dynamics of phenomena ((Dyer and Wilkins, 1991). For our longitudinal case study, we selected a single ISDP which we analyzed in detail in order to explain its dynamics and thus answering our RQs. Before we started to search for an ISDP, we established several prerequisites: (1) The requirements for the system to be developed should not yet be elicited completely. This is necessary to become aware of the analysts' perceptions of requirements uncertainty. Otherwise, it would not be possible to adequately answer RQ1. (2) Potential RE techniques should not be excluded because of the geographic distance between customer and contractor. Excluding potential techniques for this reason would make it difficult to answer RQ2 and RQ3. (3) For pragmatic reasons the ISDP should be located in Germany and scheduled for a duration between six and 18 months. We expected this period of time to be necessary to observe sufficient relevant situations with respect to our RQs. Finally, we chose a strategically important project of a leading international insurance company located in Germany that fulfilled all of our criteria (cf. section 'Research Site' for more information).

Data Collection

Overall, three researchers were involved in the data collection. In order to get an in-depth understanding of the investigated ISDP, we seek to analyze the project based on all available data sources. Multiple data sources are essential to clarify meaning by identifying different ways a phenomenon is seen (Stake, 2005). This implies an intensive data collection. At least one researcher was on-site every day during requirements elicitation phase. Data sources comprise informal interviews, observed meetings (94), project related emails (357) as well as documents (237) on the project's hard drives. Additionally, we had access to the project's RE management system. Finally, we conducted a one-day evaluation workshop with twelve participants of the first release, which was rolled out in one office. Using discussions with the whole workshop group as well as semi-structured interviews with four team leaders among the participants, we got valuable feedback to different parts of the implemented system.

Data Analysis

In interpretive research, data collection and analysis go hand in hand with each other. There is no clear separation between these two processes (Myers and Avison, 2002). Consequently, we analyzed data during data collection. The unit of analysis in our case study is a single requirements uncertainty situation, that is, a situation in which an information deficit concerning a specific requirement exists. During data analysis we had three different viewpoints, each of them addressing one of our RQs: (1) Requirements coped with and their uncertainty at different points in time, (2) RE techniques applied to cope with requirements uncertainty, and (3) success of the applied RE techniques in each observed situation. We performed data-driven coding using the software Nvivo (Bazeley, 2007). As an exploratory study, using rich insights from our interpretive case study, we did not build on existing theory (Dyer and Wilkins, 1991). Researcher triangulation was applied as at least two researchers independently coded each data piece in our database. Then, the codes were discussed and consolidated among the team.

Researchers' Involvement

In our case study, we adopt the role as neutral observers. According to Walsham (Walsham, 2006, p. 321), neutral means that "the people in the field situation do not perceive the researcher as being aligned with a particular individual or group within the organization, or being concerned with making money as consultants for example, or having strong prior views of specific people, systems or processes based on previous work in the organization."

Due to the huge support of the company, we got the status of regular employees for the time of our investigation. Thus, we were able to enter the premises whenever we wanted and got an own permanent workplace as well as an in-house exchange account. Since at least one researcher was one-site every day, we could take part in every relevant appointment including those on very short notice. In the meetings and e-meetings we took part as normal project team members with the exception that we did not make any comments regarding issues related to our research questions.

Research Site

As stated above, we conducted our case study in one of the largest insurance companies in Germany. We observed the first release of an in-house ISDP for the business department. The scope of the project was the development of an information system, in the following referred to as 'Record Management System' (RMS), which is supposed to enable the clerks to electronically view and manage their records and thus replaces the traditional physical records.

The project's first release started in March 2010 and ended in July 2011. It had a budget of about 1.5 million EUR. In total, 33 client's representatives and 15 information technology (IT) employees were involved. The client's representatives stemmed from all of the business department's four divisions and also provided the project manager. Prior to the real project, a pilot study was performed. This study's aim was to decide whether the system is going to be developed in-house or bought externally. For this purpose, main rough requirements were already elicited. However, the detailed requirements analysis phase of the subsequent in-house development project went from March 2010 until October 2010. 13 client's representatives and twelve IT employees, including four analysts, six programmers, and two architects, took part in this phase. The project manager and those four analysts will be referred to as 'project team' in the remainder of this paper. We observed the development project during the first release, that is, from March 2010 until July 2011. Figure 1 provides an overview of the project's chronicle sequence.

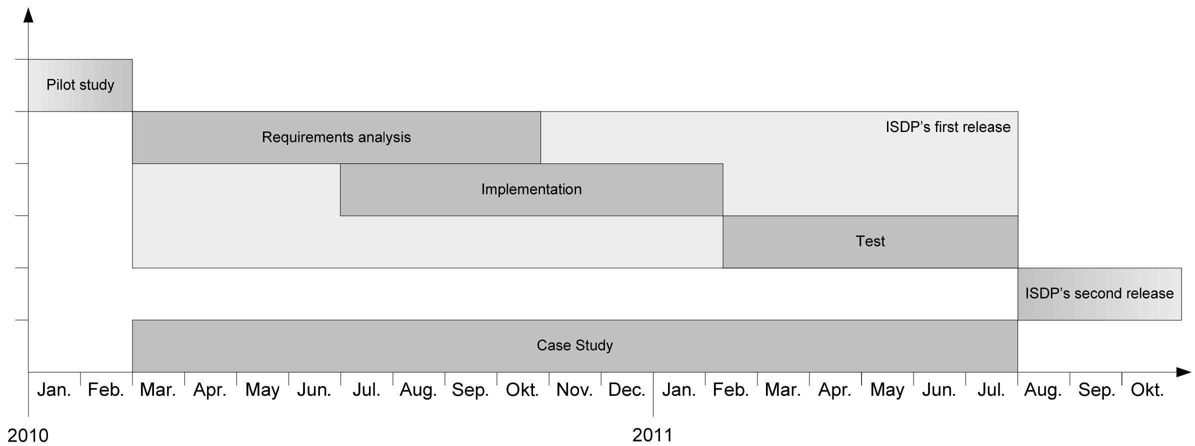

Figure 1. Overview of the project's chronicle sequence

The RMS was embedded into an existing corporate IS. It determined the screen design of the new software, which we briefly describe for a better understanding of the subsequent examples in this paper. The RMS had a threefold split screen schematically shown in Figure 2. The upper left screen, referred to as the RMS screen, shows the electronic records and their contained documents. Similar to a file explorer, records can be searched and explored and documents can be opened here. The lower left screen, referred to as the host screen, has different purposes. Here, several sub applications like a host system and a record editing tool are displayed. Finally, the right screen, referred to as the document screen, shows the opened document(s). Here, documents can be read and pages can be slightly modified (e.g., marked as out-dated, annotations and jump marks can be added).

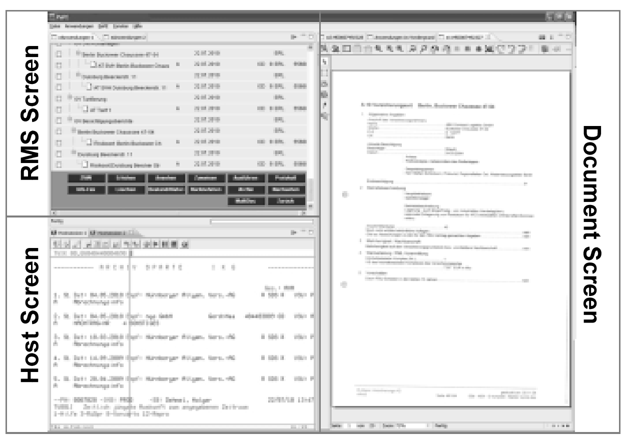

Figure 2. Schematic Representation of the Threefold Split Screen of the Insurance's Corporate IS

FINDINGS: FRAMEWORK FOR COPING WITH REQUIREMENTS UNCERTAINTY

In this section, we present a first framework for coping with requirements uncertainty on a situational level. We derive this from our longitudinal case study in an ISDP. Therefore, we perform the following four steps: First, we characterize requirements uncertainty regarding single requirements. For this purpose, we identify requirements uncertainty situation types, which needed to be coped with in the underlying ISDP. Second, we identify and differentiate RE techniques applied in the project. These techniques focus on decision making about a single uncertain requirement. Third, we evaluate the impact of the applied RE techniques for each requirements uncertainty situation type. Finally, we postulate resulting recommendations.

Requirements Uncertainty Situations Identified in Practice

In this chapter, we describe the identified dimensions of requirements uncertainty and the resulting eight requirements uncertainty situation types we use in our further analysis. For each type we give examples, which we observed in practice.

Identified Dimensions of Requirements Uncertainty

Within our case study, we observed requirements uncertainty situations. Analyzing these situations, we identified several dimensions in which the former differ from each other. However, in our further analysis, we decided to focus on specific dimensions since they only describe the characteristics of the requirement itself and not external constraints. These are: (1) number of alternatives, (2) diversity, and (3) complexity. We rate these dimensions on a high or low scale.

Number of alternatives describes the amount of possible different specifications of a single requirement. For example, the infinite number of possible sizes of a thumbnail view is rated as high. In contrast, the number of alternatives concerning the question whether the documents shall be available offline, is rated as low.

Diversity describes the extent to which the future users' needs differ regarding a single requirement. For example, situations with a lot of heterogeneous opinions among the future users or a small number of larger groups with different opinions each are rated as high diversity situations. In contrast, in situations in which most of the people agree to one alternative with just a few affected future users who prefer slightly different values, diversity is rated as low.

Complexity describes the difficulty of understanding, specifying and communicating the requirement. For example, complexity concerning the question whether all documents of an older system have to be migrated to a newer one is rated as high. Here, the user has to think about dependencies to other systems and anticipate a future workflow process. That makes it difficult to specify the requirement. We rate complexity also as high, if the question cannot be expressed in one simple sentence but needs more detailed explanation (difficulty to understand and communicate). Contrarily, the question about the size of the thumbnail view can for example be easily understood, specified, and communicated. Thus, complexity in this situation is rated as low.

Given these three dimensions, requirements uncertainty can be characterized as shown in Figure 3. Each of the three requirements uncertainty dimensions is already anchored in literature (cf. number of alternatives, e.g., Krishnan, Eppinger and Whitney, 1997; diversity, e.g., Gemino, Reich and Sauer, 2007; Wallace, Keil and Rai, 2004b; complexity, e.g., Boehm and Ross, 1989; Brooks, 1987)). By specifying our requirements uncertainty situation types as a three-dimensional construct, we also share the notion that requirements uncertainty has to be considered a profile construct (Mathiassen et al., 2007; Moynihan, 2000a, 2000b).

Resulting Requirements Uncertainty Situation Types

The three dimensions and the rating of each dimension on a high or low scale lead to eight possible requirements uncertainty situation types (see Figure 3 and Table 1). We rated each observed requirements uncertainty situation concerning the three dimensions. Thereby, we used the definitions stated above. Researcher triangulation was applied as every situation was evaluated by at least two researchers. After discussing the results, we only included situations in our further analysis with consensus among the researchers. Table 1 shows example situations. For situation types D and F we provide two examples each, since we use those in the remainder of the paper (cf. section 'Requirements Engineering Techniques Applied in Practice').

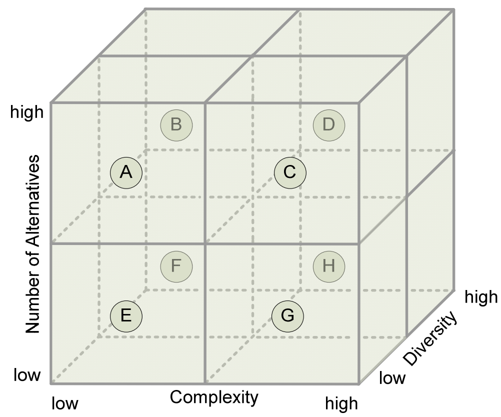

Figure 3. Requirements Uncertainty Situation Types

	Number of Alternatives	Diversity	Complexity	Given Information	Uncertainty
Type A	High	Low	Low	For each document, the number of pages shall be displayed in the RMS screen (upper left screen). It serves as an indicator for the document size.	Where shall the number of pages be displayed in the RMS screen (upper left screen)?
Type B	High	High	Low	With the record editing tool, documents can be edited. This process can also be canceled.	Shall a security mechanism be implemented to avoid cancelation by mistake?
Type C	High	Low	High	The RMS needs a 'reallocation' function. The clerk must be able to move a document from one record to another or within a record. Another branch of the company already uses a similar function which the client's representatives perceive to be close to their needs.	What distinguishes the user needs of the potential RMS users from those from the other branch concerning the reallocation function?
Type D	High	High	High	(1) An electronic record shall be searched by a set of search keys, e.g., policy number. IT prefers not to have more than 25 search keys to keep selection during search process easy. (2) When changing a record's department number, the electronic record needs to be transcoded.	(1) Which 25 search keys shall be available? (2) How shall the transcoding be performed?

Table 1. Description and Examples of Requirements Uncertainty Situation Types

Type E	Low	Low	Low	With the record editing tool, documents can be edited. It shall also be possible to combine several documents.	Shall it also be possible to combine documents of different divisions?
Type F	Low	High	Low	(1) There is a host button in the RMS screen (upper left screen), which allows users to jump into the host system in the host screen (bottom left screen). (2) There are different preferences, whether the initial host system view shall be view A or view B.	(1) What shall be the initial host system view? (2) Shall the initial host system view be view A or view B?
Type G	Low	Low	High	In the physical records, a cover page exists or the inner face of the ring binder is used for some general notes etc.	Shall a cover page also exist in the electronic record?
Type H	Low	High	High	Electronic records shall be made available offline. Thus, the clerks can take necessary documents to their customers on their mobile computers without being dependent on any connection to the insurance company's corporate network.	The following detail questions have to be answered: 1.) Shall always the complete record be available offline or shall parts of it be selectable? 2.) Shall one integrated PDF document or separated documents be generated? 3.) Shall the documents be made available in their original format in addition to the PDF? 4.) Shall the annotations be available in the offline documents? 5.) Is synchronization necessary?

Table 1. Description and Examples of Requirements Uncertainty Situation Types [continued]

Requirements Engineering Techniques Applied in Practice

In this section, we describe the identified ten RE techniques and their differentiators. According to our definition of requirements uncertainty (cf. section 'Theoretical Background'), applying an RE technique has the aim to cope with an information deficit in a decision situation with respect to one single requirement.

The basis for identifying the applied RE techniques were the observed requirements uncertainty situations. Due to several reasons (difficulties in assessing the value of the three requirements uncertainty dimensions, nontransparent application of the RE techniques, uncertain assessment of the success of the applied RE techniques due to inconsistent or unavailable statements), we finally selected 40 situations for our further analysis. For each of these situations, we analyzed the project team's behavior in detail. Then, we built groups of similar activities by comparing the 40 different situations. We identified ten distinct techniques. Finally, we analyzed the differences between the identified techniques.

This led us to five differentiators:

(1) Are the client representatives *involved* in the decision?

(2) Does the project team provide one or more *suggestions* with regard to the decision?

(3) Does the project team *present* the suggestion(s) as a mock-up, prototype, or fully implemented?

(4) Do(es) the suggestion(s) serve as basis for *further development* in cooperation with client's representatives?

(5) Does the discussion with the client's representatives take place *face-to-face*?

Table 2 shows the ten distinct techniques applied in our case study. To enable the understanding of each identified RE technique, we present one observed example for each of them in Table 3. The examples refer to one of the examples for the requirements uncertainty situation types (see Table 1). The identified RE techniques can also be found in literature (e.g. Baskerville and Pries-Heje, 2004; Benaroch et al., 2006; Beynon-Davies et al., 1999; Byrd, Cossick and Zmud, 1992; Curtis, Kellner and Over, 1992; Davidson, 2002; Davis, 1982; Galal, 2001; Hickey and Davis, 2004; Keil and Carmel, 1995; Potts, Takahashi and Anton, 1994).

	RET 1	RET 2	RET 3	RET 4	RET 5	RET 6	RET 7	RET 8	RET 9	RET 10
(1) Client's involvement	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
(2) Suggestion		No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes
(3) Presentation				No	No	No	No	Yes	Yes	Yes
(4) Further Development				No	No	Yes	Yes	No	No	Yes
(5) Face-to-face		No	Yes	No	Yes	No	Yes	No	Yes	Yes

Table 2. Differentiators of Observed Requirements Engineering Techniques (RET)

	Example Situation (cf. Table 1)	Description of Behavior	Result
RET 1 No client involvement	C	Programmers develop two different alternatives: (1) new function in RMS screen (upper left screen) or (2) integration in document screen (right screen). Project leader chooses alternative 2 due to lower effort estimation.The client's representatives are not involved in the decision.	Requirements uncertainty solved by decision of project leader. The users are satisfied with the implemented function.
RET 2 Distributed solution development	F (1)	Email with an open question of what shall be the initial host system view sent to the client's representatives.	No agreement can be reached via email. Two alternative views (view A and view B) are primarily mentioned. View A is slightly favored but not clear enough to call it a predominant opinion. Requirements uncertainty not solved through distributed solution development.

Table 3. Description and Examples of Requirements Engineering Techniques (RET)

RET 3 Common solution development	H	In a workshop, the client‘s representatives are asked detailed questions about how electronic records shall be made available offline. Solution is developed during a 20 minutes discussion within this workshop.	Detailed questions are answered. Requirements uncertainty solved by the common solution development. The users are satisfied with the implemented function.
RET 4 Forced agreement on suggestion via email	E	Project leader thinks that combining documents of different divisions is not necessary. Email with suggestion and explanation is sent to the client’s representatives with possibility to disagree within two days.	No concerns sent to the project leader. Requirements uncertainty solved by agreement of the client’s representatives on the suggestion of project leader. The users are happy with the implemented function.
RET 5 Forced agreement on suggestion in meeting	D (1)	Preselection is done by project leader. Client’s representatives are asked to agree in a workshop. However, serious discussion arises. Keys are added to the preselection and others are deleted.	Plan of preselection and nodding it through did not work out, since set of preselected search keys is modified later on. Requirements uncertainty not solved.
RET 6 Distributed solution development via email based on suggestion(s)	A	Project leader sends email with the question of where the number of pages shall be displayed in the RMS screen to the client’s representatives. She describes two suggestions in this email (1) own column (2) in existing column ‘Name’, but doesn’t explicitly state to prefer one of those two alternatives.	It was chosen to display it in an own column. Requirements uncertainty resolved by suggestions of project leader and solution development via email. The users are satisfied with the implemented function.
RET 7 Common solution development based on suggestion(s)	D (2)	As an initial suggestion, the electronic record is said to be transcoded automatically. However, within a workshop discussion, the client’s representatives decide that an automatic transcoding may be performed only if the outgoing office manually triggers the change in the host system. If this change is automatically triggered in the host system, the clerk has to perform the transcoding manually in order to be able to do some necessary changes beforehand.	Requirements uncertainty solved through common solution development in a workshop based on a suggestion. The users are satisfied with the implemented function.
RET 8 Forced agreement on presented proposal for solution via email	F (2)	Project leader sends email with a detailed presentation (screenshot and description) of one possible view, view A, to the client’s representatives. This view was slightly preferred during a first questioning. Furthermore, she asks to agree to choose this view as the initial host system view.	All representatives agree to choose view A. Requirements uncertainty solved through suggestion, presentation and request for agreement via email. The users are satisfied with the implemented function.

Table 3. Description and Examples of Requirements Engineering Techniques (RET) [continued]

RET 9 Forced agreement on presented proposal for solution in a meeting	G	In a workshop, the project leader suggests to implement a cover page. Also a possible implementation is shown and explained to the client's representatives to get their agreement on this decision.	All representatives agree to the presented solution. Requirements uncertainty solved through suggestion, presentation and request for agreement in a workshop. The users are satisfied with the implemented function.
RET 10 Common solution development based on a presented proposal for solution	B	Several suggestions for avoiding a cancelation by mistake are presented in a workshop: (1) no security mechanism, (2) security question, and (3) smaller cancel button. Pros and cons are discussed.	Security question is chosen and defined in the workshop. Requirements uncertainty solved by suggestions, presentations, and common solution development. The users are satisfied with the implemented function.

Table 3. Description and Examples of Requirements Engineering Techniques (RET) [continued]

Impact of Requirements Engineering Techniques in Different Situations

Table 4 shows the impact of the ten applied RE techniques in the 40 selected requirements uncertainty situations. We evaluated the impact on basis of our own observations and of a one-day evaluation workshop with twelve participants of the first release. Using discussions with the whole group as well as interviews with single participants, we got valuable feedback to different parts of the implemented RMS. The first digit in each cell describes the number of times the RE technique was successfully applied to the respective requirements uncertainty situation, the second digit describes the number of failures. We evaluated a situation as successful if the applied RE technique led to a decision about the uncertain requirement and the users were satisfied with the implemented requirement. We present observations concerning the different requirements uncertainty types in Table 5.

	RET 1	RET 2	RET 3	RET 4	RET 5	RET 6	RET 7	RET 8	RET 9	RET 10
Type A	1 / 0		2 / 0		1 / 0	1 / 0	3 / 0	1 / 0	2 / 0	1 / 0
Type B			1 / 0					1 / 0		3 / 0
Type C	1 / 0		0 / 0							
Type D		0 / 1			0 / 1	0 / 1	1 / 0			2 / 0
Type E	1 / 0	2 / 0	1 / 0	2 / 0					1 / 0	
Type F	0 / 1	0 / 1	2 / 0		0 / 2			1 / 0	2 / 0	
Type G			0 / 0						1 / 0	
Type H			2 / 0							

Table 4. Success and Failure of Requirements Engineering Techniques (RET)

	Number of observations	Observations
Type A	12	All techniques except 'Distributed solution development' and 'Forced agreement on suggestion via email' were applied. Every applied technique was successful.
Type B	5	The applied techniques were 'Common solution development', 'Forced agreement on presented proposal for solution via email', and 'Common solution development based on a presented proposal for solution'. Every applied technique was successful.
Type C	1	The applied technique was 'No client involvement'. This technique was successful.
Type D	6	The applied techniques were 'Distributed solution development', 'Forced agreement on suggestion in meeting', 'Distributed solution development via email based on suggestion(s)', 'Common solution development based on suggestion(s)', and 'Common solution development based on a presented proposal for solution'. The techniques 'Common solution development based on suggestion(s)' and 'Common solution development based on a presented proposal for solution' were successful, all others failed.
Type E	7	The applied techniques were 'No client involvement', 'Distributed solution development', 'Common solution development', 'Forced agreement on suggestion via email', and 'Forced agreement on presented proposal for solution in a meeting'. Every applied technique was successful.
Type F	9	The applied techniques were 'No client involvement', 'Distributed solution development', 'Common solution development', 'Forced agreement on suggestion in meeting', 'Forced agreement on presented proposal for solution via email', and 'Forced agreement on presented proposal for solution in a meeting'. The applied techniques 'Common solution development', 'Forced agreement on presented proposal for solution via email', 'Forced agreement on presented proposal for solution in a meeting' were successful, all others failed.
Type G	1	The applied technique was 'Forced agreement on presented proposal for solution in a meeting'. This technique was successful.
Type H	2	The applied technique in both cases was 'Common solution development'. Both times this technique was successful.

Table 5. Observations Concerning the Different Requirements Uncertainty Types

Recommendations for Applying Requirements Engineering Techniques

In this section, we describe the consolidated findings we derived from our observations stated above. We recommend which RE techniques should be applied in the different requirements uncertainty situation types. These recommendations should not be seen as prescriptive for every ISDP, as they stem from the observation of a specific setting - the context of the underlying ISDP, but as descriptive issues and thus basis for consideration.

In type A situations (high number of alternatives, low diversity, and low complexity) and type E situations (low, low, low), every applied technique was successful. This seems obvious concerning type E since it is the easiest situation type. Contrarily, the examples concerning situation type A show that the higher number of alternatives does not inevitably require more complex techniques including activities like making suggestions, presentations, common development, or face-to-face meetings. Thus, the analysts could select a technique on time and cost considerations. This leads to our first recommendation: *In situations of type A and E, choose an RE technique primarily based on time and cost considerations.*

In type B situations (high, high, low), techniques including presentations and/or common solution development in face-to-face meetings were successfully applied. Due to the high diversity, it seems necessary to build a consensus among future users by showing presentations or during common solution development. However, we did not observe any failures and thus cannot determine whether all those activities are really necessary. Nevertheless, this observation leads to our second recommendation: *In situations of type B, get together users with heterogeneous opinions through presentations and/or common solution development.*

In type D situations (high, high, high), just techniques with at least common solution development based on suggestion(s) were successful. If the suggestions were presented as proposal for solution, requirements uncertainty was also solved. Since this is the most challenging profile, we observed that complexity has to be reduced through suggestions and simultaneously diversity through common solution development. It seems important that solution development takes place in face-to-face meetings, so that discussion on these highly complex issues can arise among future users. Within these discussions, future users can convince others so that diversity can be reduced. Presentations seem to be irrelevant. This leads to our third recommendation: *In situations of type D, make suggestions and perform common solution development in face-to-face meetings.*

In type F situations (low, high, low), just techniques with common solution development in face-to-face meetings or a forced agreement on a presented proposal for solution were successful. Concerning the latter, it seems to be unimportant if it was performed via email or face-to-face meetings. As already discussed concerning type D situations, common solution development in face-to-face meetings induces intensive group discussions leading to mutual information exchange, shared understanding and the possibility that future users convince others by discussing pros and cons for example. Alternatively, a forced agreement on a presented proposal for solution can be used. We observed that presentations of one possible solution made it easier to convince future users to accept a solution which they initially did not prefer. This leads to our fourth recommendation: *In situations of type F, perform common solution development in face-to-face meetings or try to get the agreement on a presented proposal for solution.*

For situations of type C, G, and H, we cannot give any recommendations based on our case study because the number of observed situations and thus applied techniques is too small to understand the underlying dynamics. It seems reasonable that the number of observed situations of this type is low in this ISDP. For instance, all of them have high complexity in common which was not a prevailing characteristic of this project, as the RMS is just a replacement of physical records by electronic records.

Figure 4 summarizes our framework.

2 Identified RE Techniques to Cope with Requirements Uncertainty

RET 1	RET 2	RET 3	RET 4	RET 5	RET 6	RET 7	RET 8	RET 9	RET 10

1 Identified Requirements Uncertainty Situation Types

	Number of Alternatives	Diversity	Complexity
Type A	High	Low	Low
Type B	High	High	Low
Type C	High	Low	High
Type D	High	High	High
Type E	Low	Low	Low
Type F	Low	High	Low
Type G	Low	Low	High
Type H	Low	High	High

3 Recommendations for Applying RE Techniques

Choose an RE technique primarily based on time and cost considerations									
		✓					✓		✓
						✓			✓
Choose an RE technique primarily based on time and cost considerations									
		✓					✓	✓	

Figure 4. Framework for Coping with Requirements Uncertainty

In general, we observed that practitioners in our case study use a small set of simple techniques to cope with requirements uncertainty. Fitting the particular situation and well applied, these techniques can successfully solve any prevailing uncertainty situation concerning a single requirement. Thereby, a high number of alternatives seems to be a minor problem. Contrarily, high diversity has to be addressed by specific techniques. High diversity is best faced with common solution development in face-to-face meetings or forced agreement on a presented proposal for solution, equal if performed via email or face-to-face meetings. Concerning high complexity, we cannot make any final statement since we have only observed a high value of this dimension in connection with high diversity and thus not isolated.

DISCUSSION

In this section, we discuss our case study's lessons learned and present implications for researchers and practitioners.

Lessons Learned

This paper contributes a framework for coping with requirements uncertainty. We developed the framework based on a longitudinal interpretive case study. It consists of three parts which we consider worthwhile for future improvements in RE as an important part of managing ISDPs.

(1) *Requirements uncertainty situations.* Concerning RQ1 (*How can different situations of requirements uncertainty be characterized in practice?)*, we captured characteristics of requirements uncertainty in practice with three dimensions: Number of alternatives, diversity, and complexity. This structure represents requirements uncertainty as it was perceived as basis for decision making concerning uncertain requirements in our case study. This led us to eight situation types, characterizing typical requirements uncertainty situations in ISDPs.

(2) *Techniques to cope with requirements uncertainty.* With regard to RQ2 *(How do practitioners cope with these situations regarding the use of RE techniques?)*, we identified ten techniques applied to cope with requirements uncertainty. We structured these techniques by highlighting their differentiators. All these techniques focus on how to decide about an uncertain requirement. To answer RQ3 *(How successful are the applied techniques in coping with these situations?), w*e evaluated the impact of the applied techniques for 40 requirements uncertainty situations.

(3) *Recommendations for applying RE techniques.* Integrating the other parts (characteristics, techniques, and impact) led to recommendations about how to cope with requirements uncertainty in ISDP practice (cf. section 'Recommendations for the Use of RE Techniques in Requirements Uncertainty Situations'). For five requirements uncertainty situation types, we postulate recommendations for applying certain RE techniques. Thereby, we identified certain requirements uncertainty situation types that demand special techniques to solve requirements uncertainty adequately, that is, to make a final decision about a requirement.

The framework, especially the recommendations, can be compared to Mathiassen et al. (Mathiassen et al., 2007). Both frameworks recommend RE techniques for different situations of requirements uncertainty. The difference is that our framework focuses on decision making about single requirements, whereas Mathiassen et al. focus on the uncertainty in the whole project. Possibly, future research can combine both approaches with the aim to have one framework covering decisions on a project level as well as decisions about single requirements.

Implications

We are aware that all our observations took place in a single setting, as we conducted a single case study. Nevertheless, we believe that our findings are helpful for researchers and practitioners, as we present and explain empirically observed interdependencies between applied techniques and requirements uncertainty situation types. As our study has several limitations, we derive the following implications for future research.

Analyze situations missing in our case study: As we did not observe sufficient requirements uncertainty situations of type C (high number of alternatives, low diversity, low complexity), G (low, low, high), and H (low, high, high) in our case study, we cannot make any recommendations concerning such situations. Thus, further research should analyze these situation types to identify which RE techniques work well in such situations.

Analyze the coping with requirements uncertainty in different contexts: Our findings stem from a single case study. This allowed us a deep understanding of the coping with requirements uncertainty in the observed ISDP. The downside is that our results are limited to such a context. Other contexts may lead for example to different RE techniques, not applied in our case study, or to other interdependencies between RE techniques and their success in certain requirements situation types.

Empirically analyze requirements uncertainty on a higher level: Our study has its focus on coping with requirements uncertainty concerning single requirements. In ISDPs, requirements uncertainty also exists on a higher level, that is, affecting more than just single requirements. We cannot provide any suggestions for this issue at this point of time. Further research should also focus on these requirements uncertainty situations.

In terms of analytic generalization, we suggest practitioners to see our recommendations stated above not as prescriptive but as descriptive issues. These can be considered in requirements uncertainty situations comparable to our defined situation types. We assume that these findings may be helpful for practitioners in ISDPs, given that occurring requirements uncertainty situations are analyzed systematically. Additionally, we encourage practitioners to share their experiences in solving requirements uncertainty with colleagues. Thereby, chosen techniques in similar situations and their success can be compared. This enables colleagues and thus organizations to learn from experiences in coping with requirements uncertainty.

REFERENCES

Barki, H., Rivard, S., and Talbot, J. (2001) An integrative contingency model of software project risk management, *Journal of Management Information Systems*, 17, 4, 37–69.

Baskerville, R., and Pries-Heje, J. (2004) Short cycle time systems development, *Information Systems Journal*, 14, 3, 237–264.

Bazeley, P. (2007) Qualitative data analysis with NVivo, Sage, London.

Benaroch, M., Lichtenstein, Y., and Robinson, K. (2006) Real options in information technology risk management: An empirical validation of risk-option relationships, *MIS Quarterly*, 30, 4, 827–864.

Beynon-Davies, P., Tudhope, D., and Mackay, H. (1999) Information systems prototyping in practice, *Journal of Information Technology*, 14, 1, 107–120.

Bhattacharya, S., Krishnan, V., and Mahajan, V. (1998) Managing new product definition in highly dynamic environments, *Management Science*, 44, 11/2, 50–64.

Boehm, B. W., and Ross, R. (1989) Theory-W software project management: principles and examples, *IEEE Transactions on Software Engineering*, 15, 7, 902–916.

Brooks, F. P. (1987) No silver bullet - essence and accidents of software engineering, *IEEE Computer*, 20, 4, 10–19.

Burns, R., and Dennis, A. (1985) Selecting the appropriate application development methodology, *ACM Sigmis Database*, 17, 1, 19–23.

Byrd, T. A., Cossick, K. L., and Zmud, R. W. (1992) A synthesis of research on requirements analysis and knowledge acquisition techniques, *MIS Quarterly*, 16, 1, 117–138.

Champion, D., Stowell, F., and O'Callaghan, A. (2005) Client-led information system creation (CLIC): Navigating the gap, *Information Systems Journal*, 15, 3, 213–231.

Cheng, B. H. C., and Atlee, J. M. (2007) Research directions in requirements engineering, in L. C. Briand, and A. L. Wolf (eds.) *Proceedings of the Future on Software Engineering 2007 (FOSE '07)*, May 20 - 26, Minneapolis, MN, USA, IEEE Press, 285–303.

Curtis, B., Kellner, M. I., and Over, J. (1992) Process modeling, *Communications of the ACM*, 35, 9, 75–90.

Davidson, E. J. (2002) Technology frames and framing: A socio-cognitive investigation of requirements determination, *MIS Quarterly*, 26, 4, 329–358.

Davis, G. (1982) Strategies for information requirements determination, *IBM Systems Journal*, 21, 1, 3–30.

Dyer, W. G., and Wilkins, A. L. (1991) Better stories, not better constructs, to generate better theory, *The Academy of Management Review*, 16, 3, 613–619.

Ebert, C., and Man, J. de (2005) Requirements uncertainty: Influencing factors and concrete improvements, *Proceedings of the 27th International Conference on Software Engineering (ICSE 2005)*, May 15 - 21, St. Louis, MO, USA, ACM, 553–560.

Eisenhardt, K. M. (1989) Building theories from case study research, *The Academy of Management Review*, 14, 4, 532–550.

Fazlollahi, B., and Tanniru, M. (1991) Selecting a requirement determination methodology-contingency approach revisited, *Information & Management*, 21, 5, 291–303.

Galal, G. H. (2001) From contexts to constructs: The use of grounded theory in operationalising contingent process models, *European Journal of Information Systems*, 10, 1, 2–14.

Galbraith, J. R. (1973) Designing complex organizations, Addison-Wesley, Reading, Mass.

Geertz, C. (1973) The interpretation of cultures, Basic Books, New York.

Gemino, A., Reich, B. H., and Sauer, C. (2007) A temporal model of information technology project performance, *Journal of Management Information Systems*, 24, 3, 9–44.

Han, W. M., and Huang, S. J. (2007) An empirical analysis of risk components and performance on software projects, *Journal of Systems and Software*, 80, 1, 42–50.

Hanisch, J., and Corbitt, B. (2007) Impediments to requirements engineering during global software development, *European Journal of Information Systems*, 16, 6, 793–805.

Harris, M. L., Collins, R. W., and Hevner, A. R. (2009) Control of flexible software development under uncertainty, *Information Systems Research*, 20, 3, 400–419.

Hickey, A. M. N., and Davis, A. M. (2004) A unified model of requirements elicitation, *Journal of Management Information Systems*, 20, 4, 65–84.

Hsu, J., Chan, C., and Chen, H. (2008) The impacts of user review on software responsiveness, *Information & Management*, 45, 4, 203–210.

Jones, C. (1996) Patterns of software systems failure and success, Thompson Computer Press, Boston.

Keil, M., and Carmel, E. (1995) Customer-developer links in software development, *Communications of the ACM*, 38, 5, 33–44.

Keutel, M., and Mellis, W. (2011) Interpretive case study research: Experiences and recommendations, *MCIS 2011 Proceedings*, September 3 – 5, Limassol, Cyprus.

Krishnan, V., Eppinger, S., and Whitney, D. (1997) A model based framework to overlap product development activities, *Management Science*, 43, 4, 437–451.

Lee, A. S., and Baskerville, R. L. (2003) Generalizing generalizability in information systems research, *Information Systems Research*, 14, 3, 221–243.

Lyytinen, K., Mathiassen, L., and Ropponen, J. (1996) A framework for software risk management, *Journal of Information Technology*, 11, 4, 275–285.

Mahmood, M. (1987) Systems development methods, *MIS Quarterly*, 11, 3, 293–311.

Mathiassen, L., and Pederson, K. (2008) Managing uncertainty in organic development projects, *Communications of the Association for Information Systems*, 23, 27, 483–500.

Mathiassen, L., and Stage, J. (1990) Complexity and uncertainty in software design, in P. Dewayne (ed.) *Proceedings of the 1990 IEEE Conference on Computer Systems and Software Engineering*, October 8 - 10, Los Alamitos, CA, USA, 482–489.

Mathiassen, L., Tuunanen, T., Saarinen, T., and Rossi, M. (2007) A contingency model for requirements development, *Journal of the Association for Information Systems*, 8, 11, 569–597.

Mellis, W., Loebbecke, C., and Baskerville, R. (2010) Moderating effects of requirements uncertainty on flexible software development techniques, *5th Pre-ICIS International Research Workshop on Information Technology Project Management (IRWITPM 2010)*, December 11, St. Louis, MO, USA, 91–106.

Moynihan, T. (2000a) Coping with 'requirements-uncertainty', *Journal of Systems and Software*, 53, 2, 99–109.

Moynihan, T. (2000b) Requirements-uncertainty, *European Journal of Information Systems*, 9, 2, 82–90.

Myers, M. D., and Avison, D. (2002) An introduction to qualitative research in information systems, in M. D. Myers, and D. Avison (eds.) *Qualitative Research in Information Systems*, London, Sage, 3–12.

Na, K. S., Li, X., Simpson, J. T., and Kim, K. Y. (2004) Uncertainty profile and software project performance: a cross-national comparison, *Journal of Systems & Software*, 70, 1/2, 155.

Nidumolu, S. R. (1995) The effect of coordination and uncertainty on software project performance: Residual performance risk as an intervening variable, *Information Systems Research*, 6, 3, 191–219.

Nidumolu, S. R. (1996) Standardization, requirements uncertainty and software project performance, *Information & Management*, 31, 3, 135–150.

Orlikowski, W. J., and Baroudi, J. J. (1991) Studying information technology in organizations, *Information Systems Research*, 2, 1, 1–28.

Piekkari, R., Welch, C., and Paavilainen, E. (2009) The case study as disciplinary convention, *Organizational Research Methods*, 12, 3, 567–598.

Pitts, M. G., and Browne, G. J. (2007) Improving requirements elicitation: An empirical investigation of procedural prompts, *Information Systems Journal*, 17, 1, 89–110.

Port, D., and Bui, T. (2009) Simulating mixed agile and plan-based requirements prioritization strategies: Proof-of-concept and practical implications, *European Journal of Information Systems*, 18, 4, 317–331.

Potts, C., Takahashi, K., and Anton, A. I. (1994) Inquiry-based requirements analysis, *IEEE Software*, 11, 2, 21.

Ragin, C. C. (1997) Turning the tables, *Comparative Social Research*, 16, 1, 27–42.

Saarinen, T., and Vepsalainen, A. (1993) Managing the risks of information systems implementation, *European Journal of Information Systems*, 2, 4, 283–295.

Sommerville, I. (2011) Software engineering, Pearson, Boston.

Stake, R. E. (2005) Qualitative case studies, in N. K. Denzin, and Y. S. Lincoln (eds.) *The Sage Handbook of Qualitative Research*, Thousand Oaks, Sage, 443–466.

Stephenson, Z., and McDermid, J. (2005) Deriving architectural flexibility requirements in safety-critical systems, *IEE Proceedings Software*, 152, 4, 143–152.

Vlaar, P. W. L., van Fenema, P. C., and Tiwari, V. (2008) Cocreating understanding and value in distributed work: How members of onsite and offshore vendor teams give, make, demand, and break, *MIS Quarterly*, 32, 2, 227–255.

Wallace, L., Keil, M., and Rai, A. (2004a) How software project risk affects project performance: An investigation of the dimensions of risk and an exploratory model, *Decision Sciences*, 35, 2, 289–321.

Wallace, L., Keil, M., and Rai, A. (2004b) Understanding software project risk: A cluster analysis, *Information & Management*, 42, 1, 115–125.

Walsham, G. (1995) Interpretive case studies in IS research, *European Journal of Information Systems*, 4, 2, 74–81.

Walsham, G. (2006) Doing interpretive research, *European Journal of Information Systems*, 15, 4, 320–330.

Yin, R. K. (2009) Case study research, Sage, Newbury Park.