Available online at www.sciencedirect.com
ScienceDirect
Information and Software Technology 49 (2007) 588–604
INFORMATION AND SOFTWARE TECHNOLOGY
www.elsevier.com/locate/infsof

Requirements engineering challenges in market-driven software development – An interview study with practitioners

Lena Karlsson [a,*], Åsa G. Dahlstedt [b,1], Björn Regnell [a], Johan Natt och Dag [a], Anne Persson [b,1]

[a] Department of Communication Systems, Lund University, Box 118, 221 00 Lund, Sweden
[b] School of Humanities and Informatics, University of Skövde, Box 408, 541 28 Skövde, Sweden

Available online 13 February 2007

Abstract

Requirements engineering for market-driven software development entails special challenges. This paper presents results from an empirical study that investigates these challenges, taking a qualitative approach using interviews with fourteen employees at eight software companies and a focus group meeting with practitioners. The objective of the study is to increase the understanding of the area of market-driven requirements engineering and provide suggestions for future research by describing encountered challenges. A number of challenging issues were found, including bridging communication gaps between marketing and development, selecting the right level of process support, basing the release plan on uncertain estimates, and managing the constant flow of requirements.

Keywords: Market-driven software development; Requirements engineering; Qualitative research; Semi-structured interview

1. Introduction

This paper reports on results from an industrial qualitative survey[2], focusing on current practice and challenges for market-driven requirements engineering (RE) in Swedish software development organisations. Market-driven software can be combined with hardware in embedded systems or sold as COTS (Commercial Off-The-Shelf) products. Before problems related to inefficient RE can be properly addressed, more research is needed to better understand the challenges faced by the software industry. The purpose of the research is to discover and describe RE challenges found in industry today, in order to increase the understanding of the area of market-driven RE. The study also aims at discovering proposals for future research in the area, which is important since most available research focuses on the traditional customer-specific manner of software development. In addition, the study complements existing RE surveys, since few of them have focused on the specific challenges found in market-driven software development. The main research question investigated in this paper is: *Which RE challenges do market-driven software development companies face?*

The study focuses on market-driven software development, which is gaining increased interest in the software engineering community compared to development of customer-specific systems. This is due to the emergence of the market for COTS or packaged software [4,32]. Software is an essential part of numerous commercial products today, and hence, a growing number of companies are involved in market-driven software development. Products of all kinds, such as mobile phones, automobiles, aircrafts, toys, and games, contain software. Market-driven software products are vended on an open market and there is a large

* Corresponding author. Tel.: +46 46 222 00 00; fax: +46 46 14 58 23.
E-mail addresses: lena.karlsson@telecom.lth.se (L. Karlsson), asa.dahlstedt@his.se (Å.G. Dahlstedt), bjorn.regnell@telecom.lth.se (B. Regnell),johan.nattochdag@telecom.lth.se (J. Natt och Dag), anne.persson@his.se (A. Persson).
[1] Tel.: +46 500 44 80 00; fax: +46 500 44 83 99.
[2] By *qualitative survey* we refer to a field study including a series of interviews with practitioners.

0950-5849/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.infsof.2007.02.008

range of potential customers, thus diverse requirements need to be considered. Market-driven products are often developed in several consecutive releases for which competition is high. The characteristics of market-driven RE differs from the characteristics of customer-specific RE in several ways, as further explained in the subsequent section on related work.

There are several surveys that concern or include RE issues [5,8,12,15,18,25,27]. However, none of these focus primarily on market-driven development. Furthermore, in most of these surveys, the studied projects and organisations are mainly large, both in terms of the number of persons and requirements involved, and in terms of the duration of the projects. This qualitative survey complements the mentioned ones and provides a characterization of market-driven RE from the perspective of small- and medium sized companies, as well as fairly new ones.

In this survey, fourteen persons from eight different companies participated in interviews. After seven interviews, a short paper [24] was presented at an international workshop. In addition, a focus group meeting involving RE experts was held halfway through the study to get feedback on the challenges found so far. Semi-structured interviews were held with each interviewee. Each interview was recorded and transcribed for the analysis, which was supported by the qualitative data analysis tool Atlas.ti [35].

The paper includes a description of the companies involved, focusing on company facts, typical projects and development processes. The result of the study is a set of issues that may increase the understanding of the challenges faced by market-driven organisations, as well as indicate the direction of further research. In addition, we discuss our experiences from using a qualitative research approach.

The remainder of the paper is organised as follows. In Section 2, related work is presented. The research method is described in Section 3. Section 4 presents the results of the study and ends with a summary. Section 5 discusses the results and relates the findings to earlier work. It also discusses the threats to validity and our research experiences. Section 6 concludes the paper and presents some ideas for further work.

2. Related work

This section describes some earlier work related to the topics of market-driven RE and RE surveys. The included references are the ones found most important to our study, either with regard to the research design or to the research results.

2.1. Market-driven requirements engineering

Although there are many commonalities between market-driven and customer-specific development, the literature argues that there are also differences of such kind that these need to be made explicit [34]. The major differences include the characteristics of stakeholding and schedule constraints [32] as well as release planning and managing the constant flow of new requirements [4,19,30].

In market-driven projects, there is no distinct and defined set of users [32]. There are mainly potential users, an imagined group of people who may fit into the profile of an intended product user. Eliciting requirements from this group of users and customers is one of the major distinguishing characteristics between market-driven and customer-specific RE [10,30,33]. This is mainly managed through marketing, technical support, user groups and trade publication reviewers [4].

Often, requirements are *invented* by the developers [30], e.g., based on strategic business objectives, domain knowledge and a product vision. The development organisation is the primary stakeholder, and hence it decides which requirements to use in the next release. Nevertheless, in order to keep, or increase, market shares, the requirements that satisfy most customers need to be selected. This further emphasises the role of marketing in the market-driven development situation [10].

Within market-driven development organisations, *time-to-market* is described as a survival attribute [28,33]. If the product is not released to the market on time, there is a risk of loosing market shares to competitors. As a consequence, the release dates are kept fixed and requirements of a lower priority may hence be excluded from the current release in case of a delay [2]. Release planning, aiming to select a set of requirements for the next release that maximises customer value taking into account the available resources [3], is therefore a major challenge [33]. However, there is usually a steady stream of new requirements, improvements suggestions, complaints and bug reports suggested by existing customers and users of previous releases. Therefore, effective prioritisation and cost/impact assessment is needed to support the release planning task [2,23,33]. Moreover, the RE process needs to include procedures to capture and preserve this steady stream of requirements [16].

2.2. Requirements engineering surveys

There are several surveys that concern or include RE related challenges. The classical article by Curtis et al. [8] mainly involves large system development projects, both customer-specific and market-driven ones. Although their survey does not focus solely on RE, it identifies three major challenges which all can be related to RE: the thin spread of application domain knowledge, fluctuating and conflicting requirements, and communication and coordination breakdowns. A few years later, Lubars et al. [25] published their field study on requirements modelling. Their study also includes both customer-specific and market-driven projects, as is also the case in the paper by Curtis et al. [8]. The challenges found in [25] is well in line with the ones in [8] and include e.g., vaguely stated requirements, missing requirements, requirements misunderstandings, the lack of

definitive requirements from outside the development organisation, lack of customer contact, and frequently changing requirements.

Next, two studies on RE practices are published, which both also report on experienced problems [5,12]. In the paper by El Emam and Madhavji [12], the challenges identified are more related to project management issues such as performing the appropriate activities, deciding when to stop particular activities, ensuring an adequate level of user participation, and selecting capable personnel for the key roles in RE. The survey by Chatzoglou [5] mainly includes projects of a customer-specific characteristic. The challenges presented are e.g., lack of enough resources for a successful completion of the RE process, and not adequate quality of tools and techniques employed in the RE process.

Later on a study on RE challenges was presented by Kamsties et al. [21]. Unlike the field study by Curtis et al. [8], this study includes small- and medium-sized enterprises. However, both customer-specific and market-driven projects were represented. Kamsties et al. [21] agree to some extent with Curtis et al. [8] and Lubars et al. [25] and identify unclear and incomplete requirements as a common challenge. Other challenges mentioned are: complexity of requirements documents, lack of documented requirements in market-driven projects, and cases when implementation of new requirements can cause unforeseeable interaction with requirements already implemented.

Two more recent papers on RE challenges are written by Hofmann and Lehner [18] and Hall et al. [15]. The first survey aims at establishing a clear link between RE practices and performance, and includes both customer-specific and market-driven projects. Also in this paper, changing requirements are identified as challenging. They also support the findings by El Emam and Madhavji [12], concerning selecting capable personnel, since they found that there is a lack of involvement of technically knowledgeable stakeholders when defining the initial requirements. Other challenges identified in [18] are unrealistic requirements posed by marketing, and perceived ad hoc RE processes. Moreover, they have also shown that approximately 15% of the project effort was spent on RE activities. The paper by Hall et al. [15] emphasises that most requirements problems are organisational rather than technical. Hall et al. also discuss a relation between the maturity of the companies and requirements problems found.

Finally, there are a number of RE surveys that are not focusing on challenges [1,27] and others which investigates certain kinds of challenges, e.g., the survey by Damian and Zowghi [9] which describes challenges caused by stakeholders' geographical distance.

In general, none of the presented RE surveys have a primary focus on market-driven development, although some includes both market-driven and customer-specific organisations/projects [8,18,21,25]. Hence, none of these gives an overview of challenges faced specifically by market-driven development organisation regarding RE issues, despite the fact that there are pronounced differences between customer-specific and market-driven RE. Therefore, there is a need for an investigation of the RE challenges faced by market-driven development organisations.

3. Research methodology

The study was carried out using a qualitative research approach. Qualitative research aims to investigate and understand social and cultural phenomena in the context where they exist. Furthermore, qualitative research aims to place the studied phenomena in a holistic framework and suggests that social and cultural phenomena are best investigated by studying people's actions in and verbalised thoughts about the social context in which they act [26]. Qualitative research methods are useful when the purpose is to explore an area of interest, to obtain an overview of a complex area, and to discover diversities and variety rather than similarities [31]. It is also preferable to use a qualitative approach when the aim is to improve the understanding of a phenomenon where little is known. This is due to the fact that it focuses on gaining in-depth information [17].

The purpose of the study presented in this paper is to gain in-depth understanding of the nature of requirements engineering within market-driven software companies, specifically focusing on exploring the challenges software development companies in this sector face. The aim is also to provide a basis for formulating hypotheses for future research. Due to this explorative nature of the study a qualitative approach has been considered suitable.

The study is mainly built on semi-structured interviews with a high degree of discussion between the interviewer and the interviewee, complemented with a focus group meeting. This approach has enabled the researchers to gain in-depth understanding e.g., of the concepts/terminology used by the companies. This proved to substantially facilitate data analysis. For example, concepts such as requirement, process and method had different meanings among the companies. An alternative approach could have been to use a questionnaire. The drawbacks of such an approach would be that the assumptions of the researchers in that case govern the questions asked and the terminology used. The explorative element of the study would, hence, be difficult to achieve. Furthermore, a questionnaire does not provide the opportunity to discuss what questions and concepts mean and to explore new views on the area that may appear during a face-to-face interview.

Quantitative methods are sometimes claimed to be "better" than qualitative methods with the argument that they provide objective measurement and enable replication of studies, as opposed to qualitative methods that build on inherently subjective interpretation. However, we do not consider debating which type of method is "better" than the other to contribute constructively to science. Instead we need to take a more general view of long-term research (Fig. 1). The cycle of long-term research entails (1) discov-

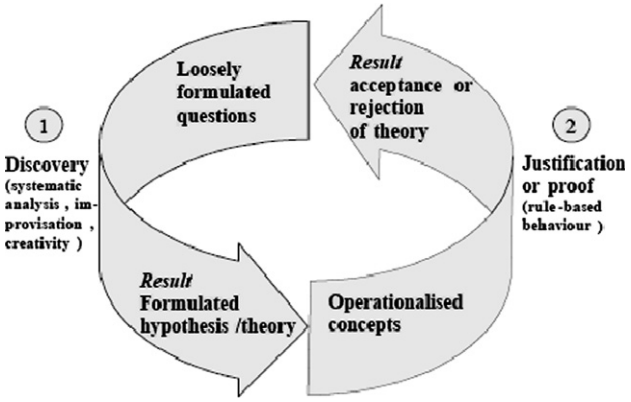

Fig. 1. The long-term cycle of research.

ery of hypotheses/theories and (2) justification/proof/validation of hypotheses/theories. The discovery phase is followed by justification and then new theories or elements of theories can be discovered, which need justification, etc.

In the study reported in this paper, we have discovered a number of challenges to RE in market-driven software development companies. These challenges can be seen as hypotheses or be the basis on which new hypothesis, studies and experiments can be formulated and carried out.

The study consists of three steps, which are described in the following. Each step is divided into three phases: planning, data collection and analysis (Fig. 2).

3.1. Step 1 – Interview study (part 1)

3.1.1. Planning

The first part of the study involved a brainstorming and planning meeting to identify different areas of interest and to plan the study. We used a combination of maximum variation sampling and convenience sampling since we selected companies with as different characteristics as possible within our industrial collaboration network [29]. We aimed at a variety of companies, with respect to size, type of process, application domain and type of product. We started with five companies, in which seven persons were interviewed. Interviewees in different roles were involved in order to collect different viewpoints and perspectives on the nature of RE.

The interview instrument was designed with respect to the different areas of interest and with inspiration from other RE surveys [12,25]. To test the interview instrument, two pilot interviews were carried out. Some questions were clarified and the structure of the interview instrument was improved before interviewing proceeded. A summary of the interview instrument is available in the appendix, Table A.

3.1.2. Data collection

The study uses a semi-structured interview strategy [31]. All interviews were attended by one interviewee and three interviewers, one of which was responsible for the interview process. The other two took extensive notes in order to

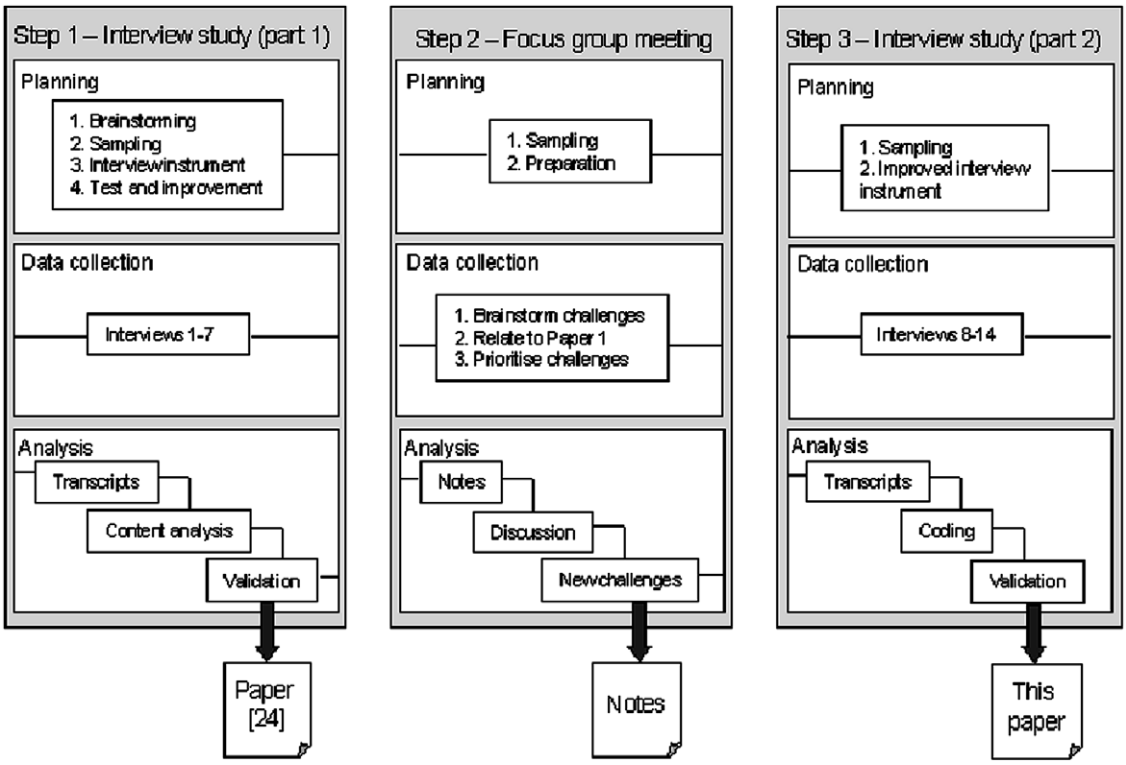

Fig. 2. The research procedure used in the interview study.

gather as much information as possible. All interviews, varying in length from 90 to 150 min, were also recorded. Transcripts of all interviews were made in order to facilitate and improve the analysis process. The transcripts varied between 7 and 23 pages in size.

3.1.3. Analysis

The *content analysis* [29] involved marking and discussing interesting sections in the transcripts. The interviewers examined the transcripts from different perspectives and searched for explicitly stated or concealed RE challenges. Another researcher, who did not attend the interviews, also analysed the transcripts to enhance validity. The results from Step 1 are presented in [24].

3.2. Step 2 – Focus group meeting

3.2.1. Planning

The aim of the focus group meeting was to obtain feedback from industrial RE experts on the eleven RE challenges found so far, and to find additional ones. Therefore, we selected two participants who were interviewed in Step 1, and three new participants that did not attend the interviews.

3.2.2. Data collection

The meeting started with a brainstorming session where the five industrial RE experts wrote down challenges from their own experience on post-it notes. The session was facilitated by one of the five researchers present. Next, the industrial experts mapped their post-it notes to the challenges found in Step 1. Some post-it notes could not be mapped to a challenge and therefore three new challenges were formulated. All fourteen challenges were prioritised based on a simple scheme; every participant had ten votes each to distribute among the challenges they found most important.

3.2.3. Analysis

The five researchers took extensive notes, which were discussed and composed after the meeting. No particular analysis was made of the notes at this stage. Instead, the notes were used as input to the analysis of the third step, the second part of the interview study.

3.3. Step 3 – Interview study (part 2)

3.3.1. Planning

We continued with the sampling strategy developed in the initial step of the study, and interviewed seven persons at five software companies. The new companies included both large and small organisations[3]. Both very large and very small projects appear in the new companies and more attention was also paid to developers of embedded systems in order to broaden the scope.

The interview instrument was adjusted. Some questions were enhanced with more detail, while others were given a more open-ended structure.

3.3.2. Data collection

The semi-structured interview approach was continued. The interviews varied between 60 and 120 min in length. One interview was attended by two interviewees due to lack of time. All interviews were conducted by one to three interviewers and were recorded on tape. Notes were taken in the same manner as during step 1 and each interview was transcribed before analysis.

3.3.3. Final analysis of all data

Since we sought a comprehensive view of the complete data set, the data from the first part of the study was reanalysed together with the data from the focus group meeting and the second part of the interview study. The analysis was supported by the qualitative data analysis tool, Atlas.ti [35]. As the data grew rather extensive (more than 160 pages of transcribed text), tool support was necessary in order to maintain structure, as well as to facilitate cooperation between the researchers, who worked at different geographical locations.

In the final analysis we used the eleven challenges that emerged in the first set of interviews together with the additional three that were discovered during the focus group meeting as predefined categories/codes [14]. In addition, one more category emerged as our understanding of the data increased through discussing and working with the data. Thus, a total number of 15 categories were found (Table 1). Each category was described with quotations

Table 1
The 15 categories used during analysis

Challenges that emerged from the interview study (part 1) [24]	Challenges that were added after the focus group meeting	Challenges that emerged from the final analysis of all data
Design in the requirements	Requirements for requirements	Non-functional requirements
Simple techniques and tools	Organisational culture	
Requirements repository	Tool integration	
Requirements bundling		
Requirements overload		
Requirements changes		
Market-driven/ technology-driven		
Organisational stability		
Process		
Release planning		
Gap between marketing and developers		

[3] Due to lack of organizational information regarding annual turnover our definition of small-, medium-, and large organizations is approximated. For an elaborated definition, see e.g., [http://ec.europa.eu/enterprise/enterprise_policy/sme_definition/sme_user_guide.pdf].

and examples, in order to develop a common understanding of each category. Within each category we found more detailed challenges, which were further analysed.

With reference to Grounded Theory [14] the process throughout the three steps of the study reached what can be described as theoretical saturation. The number of new categories that emerged from each step became smaller and smaller, from eleven to one. Apart from one more category, the third step mainly provided more detail and confirmed previous findings. This is illustrated in Table 1.

The interview transcripts were divided between two researchers for coding. The transcripts were analysed and interesting quotations were marked with one or more of the 15 categories. Afterwards, the researchers switched interviews to validate that coding was made in a consistent manner. It was also possible to add more categories at that point. Each researcher kept a file in the tool that contained the coding made by that researcher. When coding was finished, the two individual files were merged into one within the analysis tool. For the analysis, all related transcript quotations for each category were compiled and printed in order to compile the data into a readable format. The results from the analysis are found in Section 4.

In order to further ensure the quality of the coding process, a third researcher that did not attend the interviews compared the transcripts with the results given in this paper. Quotations in the results were traced to their source in the transcripts to check translation and to ensure that the interpretations converged. Due to traceability problems, caused mainly by translation from Swedish, two quotations were not recovered in the extensive transcript material and were therefore removed from the results. One quotation was changed to a descriptive explanation, since it was difficult to translate without losing the essence of what had been said. In six quotations, single words were erased or changed to obtain a better language structure.

4. Results

This section presents the challenges discovered during the analysis of the interviews and the focus group meeting. The twelve following sub-sections present and discuss one or several challenges corresponding to one or several of the codes presented in Table 1. Company and interviewee characteristics are available in the appendix, Table B. The challenges are subsequently elaborated and explained using excerpts from transcripts and interpretations from coding and analysis.

4.1. Simple techniques for basic needs

The Rational Unified Process (RUP) [20] was mentioned by several of the interviewees as being the development process used. For example, the employees at Company F have used RUP and Rational tools for a few years and they are generally satisfied with their performance. However, some problems were mentioned, e.g., that RUP lacks support in the handover between requirements engineers and designers. Another issue is that RUP is a use case driven approach, while the company wanted to work in a feature-driven manner. This adds some workload as the traceability between features and use cases needs to be maintained.

At Company E, RUP and Rational tools were tested, but the employees had trouble accepting the process. *''It just gets too complex''* was the comment from the managing director. At Company B, RUP was under consideration but it was difficult to find the necessary time and effort for it to be introduced in the organisation. The project manager's comment was *''We do not have time to implement RUP and those tools''*. The product manager at the same company said that they want a simple tool, *''that you do not need a thick manual to understand''*. The project manager requested an RE tool, such as DOORS, to support RE in the projects. Thus, the need for tool support seems to vary depending on organisational role. It also seems to be the smaller organisations that have trouble finding processes and tools that suit their needs. During the focus group meeting, a lack of tools that support a flexible process was discussed. The difficulty to integrate different tools (e.g., RE tools and testing tools) was also an issue.

Some companies use the Unified Modelling Language (UML) [13] for specification. The opinions about UML vary among the interviewees. At Company C, UML is regarded as easy for customers to understand, while at Company E it is stated that *''you cannot take the models to someone without a masters degree''* and therefore it is not a good means for communicating with customers. The difference in views regarding UML is probably due to different customer characteristics. The software developed by Company C is intended for developers and product managers, while the software developed by Company E is intended for managers and developers of business processes. Company H has tried a use case methodology but it was abandoned since they perceived that it had shortcoming as a feedback instrument in customer communication. On the other hand, they had positive experiences with prototyping to get customer feedback. They stated that they need techniques and methods that are easy to understand in order to get customer feedback.

4.2. Communication gap between marketing staff and developers

At the focus group meeting, the lack of co-operation between different parts of the organisation, especially marketing and development, was recognised by most participants. One participant stated, *''If these two departments communicated, we would not even have to specify the requirements. [...] If everyone has the same goal and vision, then everyone works in the right direction''*. At Company B, the project manager suggests that one way to solve the lack of communication between marketing and developers is

for marketing staff to learn UML as it could function as a common language for communication of requirements.

In Company A, the communication gap between marketing and development was also obvious. The marketing department's view of specifying requirements was to write down ideas for future functionality, while the developers expected clear and detailed requirements that could be used for design. Nobody took responsibility for the specification and analysis of requirements. At Company F, the problem was managed by a systems management group, which acts as a "mediator" between marketing and developers as well as re-formulates the one-line requirements from marketing into designable requirements for the developers. This is a good solution according to the system manager at Company F, as product management does not have as detailed knowledge as systems management.

Another communication problem concerns providing the sales department with sufficient information before customer meetings. At Company B it had been the case that sales staff promised functionality to customers before confirming its feasibility. At the focus group meeting this was mentioned as a challenge and it was suggested to have the sales department communicate with requirements engineers to solve the problem.

Another indication of communication gaps between departments is the different opinions as to what constitutes a "good requirement". At Company B, the product manager thinks that a good requirement is one that yields high revenue with low effort, and that is possible to sell. The managing director at Company C expresses a similar view. However, according to the developers in the same organisations, good requirements are independent, testable, clear, and not conflicting. It seems that the notion of a "good requirement" vary depending on organisational role.

4.3. Writing understandable requirements

Most of the companies use natural language (NL) to document their requirements. This is sufficient according to most interviewees since it is the customers' language and customers need to understand the requirements. However, one of the interviewees at Company H states that different stakeholders use different vocabulary, or *"they use the same word but with a different meaning"*. This makes NL problematic.

Producing well-formulated requirements is an issue, since many of the interviewees find it difficult to understand requirements. The head of developers at Company C considers it as one of the major challenges in carrying out his daily tasks. However, the managing director at the same company does not consider this to be a major problem. The difference in views seems to be related to the role within the organisation. It is possible that marketing staff do not need the deeper understanding of requirements, and can therefore cope with less well formulated requirements. However, developers may need more detail to manage implementation. Many interviewees put effort into increasing the clarity and understandability of requirements, mainly through group discussions. The head of developers at Company C said, *"It is not possible to write requirements completely, clear as a bell. You always need to discuss with others"*.

4.4. Managing the constant flow of new requirements

Market-driven product developers need to deal with a constant flow of new product requirements. It is considered important to collect all these new and potentially valuable ideas, derived from e.g., customers, users, developers, and support personnel. However, there needs to be efficient means to manage them. Several of the companies use a repository or tool to store incoming requests and these tools are then used to search for interesting, and valuable requirements to implement in the next release of the product. In Company B, C, and G, the same repository is used for requirements and trouble reports (TR), since both compete for the same resources. Even though a repository ensures that no good ideas are lost, it poses a number of other challenges.

First, Company A, C, and D allow their customers and other stakeholders to enter requirements and ideas by themselves. However, this requires various efforts afterwards in order to improve the understanding of the statements and to identify duplicates amongst the requirements.

Second, Company A, which has a mature product, has had severe problems with overload in the requirements database, since more requirements are added than could be managed. This has resulted in difficulties when prioritising the requirements for the next release, since there were thousands of requirements to consider. The problem was temporarily solved by creating a list that included requirements considered the most important for the next release. There is, however, an apparent risk that some rejected requirements would have been more important after all.

Third, the project manager at Company D emphasises the importance of giving feedback to all suggestions from stakeholders. Otherwise they may feel neglected and *"if they knew [their suggestion] was sent into nowhere, they would stop suggesting"*. Thus, it is highly important that customers know that all suggestions are taken into consideration. However, giving feedback to the constant flow of new requirements from a large number of customers requires a large amount of effort.

4.5. Requirements volatility

Requirements change for a number of different reasons such as e.g., when the market fluctuates, problems are found during coding or reviews, and resource constraints change. Some interviewees fear volatility, while others welcome it. For example, at Company B the project manager said, *"it takes time before the product is stable, and that is what you want to reach"*. At Company D on the other hand, volatility is accepted as a fact of software develop-

ment. The project manager states, *"[The customers] will change their minds when they see it, they change their mind in any case"*. Therefore they use an agile development approach. The project manager also said that *"Code is [...] cheap to throw away, [...] it's almost for free"*. Therefore it is possible to build imperfect code and show it to the customer for feedback. This approach is also used at Company C, where a function is often developed to 60–90% so that feedback can be received earlier without having to spend 100% of the planned resources. The developer said, *"it's not worth finishing a requirement to 100% because change requests will come [...] so we accept it earlier and wait for comments"*. The trade-off between stability and volatility was also discussed at the focus group meeting where one participant said: *"the environment changes while we want to freeze the specification and strive for stability in the projects"*. Another participant requested requirements models that are easy to change.

4.6. Requirements traceability and interdependencies

All participants involved in the study acknowledge the existence of relationships and interdependencies among requirements. However, their perceptions with regard to its importance and impact on the development work differ substantially between companies. Influencing factors seem to be how elaborate and well-defined the RE process is, but more importantly the size of the project and the complexity of the product.

In six of the companies, relationships and dependencies between requirements are usually used for bundling related requirements to be implemented together. Bundles are made in order to increase efficiency during implementation, e.g., because the requirements relate to the same part of the system or should be implemented by the same developer.

Company F has a more rigorous way of dealing with requirements interdependencies. They use a so-called *anatomy plan* to describe the functional structure of the software. It is a description of how the functionality of the system is related, i.e., dependencies between functions. It is mainly used to describe the order in which the functionality of the system should be implemented. Company F states that even if it is difficult, functional dependencies can be managed in this way. A major problem is to deal with what they call *dynamic interdependencies*, i.e., quality characteristics or non-functional requirements, which influence a larger part of the functionality or other characteristics of the system. The problem is not only to know that the interdependency exists but also to figure out *how* the requirements affect each other and how this can be dealt with.

According to both Company C and D, *duplicates* amongst requirements are problematic. Often, the same idea or solution comes up several times, but takes different shapes. In addition, these could be mutually exclusive, i.e., they cannot exist at the same time within the system e.g., since they are alternative solutions to the same idea.

4.7. Requirements are invented rather than discovered

Most of the interviewed companies express a trade-off between their own innovative requirements and the requirements suggested by customers and users. At both Company B and C, approximately half of the requirements are suggested by users and half are internally *invented*. At Company B, both interviewees express a wish to be more market-driven. *"That is alpha and omega when creating a product success, to run it from marketing"*, as stated by the project manager, continuing *"R&D can always create cool things, [...] and they think of themselves as ordinary users"*. As engineers tend to be early adopters of new technology, developers may not represent the ordinary market for these high technology products.

One of the risks acknowledged by Company B is that when a technology-focused company is founded there is no need to think about who the customers are; it is enough to have some new piece of technology. This is a dangerous situation since, eventually, the company needs to find a customer base and adapt the product to actual customer needs. There is also a need to find the right distributors and retailers. *"A product may be incredibly good, but in the wrong store it will still not be sold"*, is the comment from the project manager at Company B. The same person also states that in a new technology-focused company, *"the developers are the heroes"* and therefore marketing has trouble influencing the product functionality. At the same time it is understandable that customer input is difficult to obtain when the product is unique and not known by users. The market simply does not know what to wish for.

At the focus group meeting several participants expressed problems with getting the necessary market input, *"[...] those teams consist of happy engineers who invent requirements rather than listen to me"* as one of the participants stated. The focus group participants also discussed non-functional requirements (NFR). One participant stated that NFRs often are suggested by customers, while functional requirements often are internal. This could also reflect the fact that many companies are technology-focused and that functional issues therefore are perceived to be more important than non-functional issues.

Although the interviewed organisations are all in the market-driven area, several of them have distributors and retailers who sell the products and it can therefore be difficult to reach end-users for feedback. Thus, both retailers and end-users need to be considered during requirements analysis. In Company A and C, the number of customers is fairly limited, and some large customers have more influence than others. This is also acknowledged by the marketing person at Company E, who describes problems with getting the functionality generic enough to suit several customers.

4.8. Implementing and improving RE within the organisation

A major challenge is to get acceptance for new ways of working with RE, i.e., to implement changes to current

practices within an organisation. It includes changing the behaviour of people. Succeeding with the change process is, however, crucial in order to improve RE. A focus group participant stated: *"The change process is the most difficult – to change the way people work. There is a lot of good ideas, tools and methods, but it is too tedious to start using them"*. Both interviewees at Company F state that the biggest challenge is to make everyone follow the same process and work in the same manner.

It is important to realise that it takes time to implement a new way of working, and education of the employees is required. To make a group of developers embrace a certain way of working requires a high level of confidence in the person responsible for the implementation of the new approach. Company F has defined a role called *process engineer*, who is responsible for implementing process changes and educate employees. The process engineer works together with the project as a support function in matters related to how the process should be carried out. The process engineering also puts effort into *pre-phase training* of employees in order to improve their ability to understand and use the defined process.

Interviewees from both Company A and B state that the project manager controls how the process is carried out in practice, and therefore the process may vary between projects. It is necessary to obtain acceptance for putting effort into RE amongst project managers before implementing changes in RE practices, i.e., the ideas of improving RE must be well supported by the actors who control the way of working within the organisation. Some participants at the focus group meeting highlighted the issue of transferring knowledge between projects. It is difficult to make people learn from success stories, and apply this knowledge in other projects, especially in organisations where the acceptance for putting effort into RE is low. One focus group participant stated, *"All projects where RE has been used, have been successful. However, it is not certain that project managers who have used RE in one project use it in the next one"*.

4.9. Resource allocation to RE

One major problem, emphasized during the focus group meeting, is the lack of resources to conduct RE. Participants asked for example: *"How do you get time and resources to the requirements work?"*. The problem was explained by a lack of understanding for the time and resources needed to carry out high quality RE, e.g., amongst management. Lack of resources for RE may cause many of the other problems discussed during the meeting, e.g., badly formulated requirements.

Both Company A and B have experienced problems with obtaining enough resources for RE. Often, implementation begins as soon as the project starts, i.e., it is carried out in parallel with the RE work. As a consequence, the ability to put effort into RE is very low, especially since key personnel are caught up in other projects. Interviewees at Company A and B express a need to start and finish the RE work before implementation begins.

Most of the companies estimate that they spend approximately 10–15% of the total effort on RE, although none of the companies actually measure it. However, their opinions regarding whether this is enough or not differ. Company A and B state that too little effort is spent on RE and that much could be improved in their RE practices, while Company C and D consider it to be just the right amount of effort.

4.10. Organisational stability

Software engineering is a knowledge intensive activity, and is naturally dependent on the development staff's knowledge about the product, its usage environment, and their knowledge about the development process. The competence of involved staff is critical when the process is ad hoc. The project manager at Company B said: *"Everybody knows everybody, and everybody helps everybody. That is the reason a small company can survive with an ad hoc process"* and *"The projects depend on the individuals. Some projects would have died if some people would have left"*. Of course, it is risky to be too dependent on individuals, a problem that became apparent when the company downsized and lost part of its staff. Competence and knowledge was lost. Company C and D also discussed the knowledge capital in the staff and emphasize that their success is thanks to the employees.

Other organisational issues such as co-ordination and communication are key issues within the organisations involved in the study. Company F describes problems with deciding how to organise projects in order to make co-ordination and communication as efficient as possible. Larger projects enhance the overview but complicate co-ordination and they are difficult to manage. Smaller projects facilitate communication within the project group, but make it more difficult to create a comprehensive, overall picture. The systems manager stated, *"It is always a matter of how you choose to divide, so to speak. Either you get problems with communication or you get problems with getting the overall picture right"*.

4.11. Selecting the right process

Several of the interviewed companies struggle with implementing, evolving, or adjusting their development process. Company B is about to develop and implement a development process as the lack of structure in the current way of working is considered to be a major problem. The project manager said: *"Yes, I believe that there is an infinite amount of drawbacks with our current way [of working]"*. The process is a way for project managers to communicate and interact with the different parties involved in a project. Both interviewees at Company B also emphasised that, to start with, only the most important elements should be included in the process. It is easier to evolve the

process gradually, rather than trying to do everything at the same time.

Company A, used to had a very detailed process, but it was slimmed down when the product became more mature and fewer changes were made in each release. Unfortunately, it became too slim, which resulted in problems related to lack of structure and control. Therefore, they are still struggling with defining a process on a suitable level of detail.

Company C and D were generally pleased with their way of working. Even though the processes were not rigorous, they had control of their requirements and use tools to structure and maintain the requirements. Both these organisations are fairly small and may therefore have less need for structure and processes.

Company F has put a lot of effort into constructing a specific development process that everyone involved follow. However, they emphasised the importance of making the process flexible enough to be adjustable due to specific situations. Using process engineers is one way of achieving a good interaction between projects and process, and to facilitate the evolution of the process within the projects.

When adjusting the development process, other support processes, such as project management processes need to be adjusted as well. Company F had experienced some problems with combining the iterative RUP process with the sequential toll gate (TG) [7] model demanded by management. Certain business decisions require certain information, which is not produced until later iterations. The systems manager said: *''the planning of each iteration cannot be finished at TG2 because then the RE work is not iterative''*. In order to ensure that these processes are aligned and support each other, it may be necessary to adjust them, e.g., by redefining the business decisions stipulated by the TG process.

4.12. Release planning based on uncertain estimates

Release planning requires consideration of many different aspects. Most of the companies had some cost-value approach, and consider aspects such as e.g., customer or user value, development cost, strategic value, and marketing positioning in their plan. There are also other aspects, such as effects on the architecture, implementation risks, interdependencies, and deadlines. For embedded systems, even more factors are added, such as component size and availability, as well as sales volumes.

The interviewees at Company F stressed the importance of considering the hardware and non-functional requirements before applications, as architectural aspects may change the structure of the system. These interrelationships are visible in their ''anatomy plan'', which means that interrelated use cases and functionality can be developed in the same release. The systems manager said: *''We want the requirements in one iteration to be connected so that we do not have to change the same documents during the next iteration''*.

Requirements prioritisation was discussed by all participants in the study. Several companies tended to use some sort of numeral assignment [22] to arrange requirements into groups. Requirements are often divided into high, medium and low priority, or given a number between 1 and 5. At Company A, the project manager constructed a top-10 list including the top priority requirements. Another 10 requirements were regarded as medium priority and were put below in the list. Other companies, such as Company C, had a more rigorous approach and used a requirements management tool for prioritisation. At Company B, requirements prioritisation was more ad hoc, or as the project manager said when discussing different stakeholders: *''the one who shouts the loudest wins''*.

Another release planning issue regards effort estimates. Several of the interviewees expressed that it is an important task. Yet, none of the interviewees mentioned any particular method for estimating effort. At Company C the importance of effort estimates during release planning was stressed: *''It is important that the requirements end up in the right release and if we estimate incorrectly it will affect the release plan''*. At Company G, interrelated requirements were sometimes grouped and estimated if they were to be implemented together, because it could be easier than estimating each requirement separately. The opposite opinion was expressed by the project manager at Company D who said that smaller requirements are easier to estimate correctly. Company F usually estimated effort per use case instead of per feature because each feature could affect several use cases. At Company C, the developer said that it is necessary to have detailed knowledge of the system in order to know which parts of the code that is affected by a new requirement.

At Company C and D, effort estimates were discussed between project members so that contradicting estimates were resolved. The developer at Company C said: *''we know that we have done it correctly when everyone agrees on the estimate''*, and at Company D, the project manager said that *''we get better estimates if people have to explain how they think''*. Both Company C and D had weekly meetings, where requirements were discussed, effort estimated and release planned. At Company A, meetings were held with marketing personnel and developers so that those who suggested requirements, i.e., marketing personnel, can explain the purpose to the developers. The developers found it easier to estimate the development effort for requirements when they understood the purpose. The product manager said, *''High-level requirements are difficult for developers to estimate''*.

4.13. Summary of findings

This section summarises and discusses the challenging areas presented above. Table 2 provides a list of the discovered challenges together with a summary of some of the problematic issues, called *key problem areas*. Although the focus is on market-driven organisations, some of the

findings are of more universal relevance. Issues related to communication and coordination may appear also in customer-specific organisations, as well as issues regarding requirements traceability and interdependencies. However, five of the challenges are directly related to the market-driven characteristics of the participating organisations. Those challenges are marked in Table 2 and discussed below.

The special stakeholder characteristics, i.e., several potential customers and users on a large and open market, contribute to some of the challenges. For example, the constant flow of requirements (Section 4.4) is caused by the variety of stakeholders who have demands on the product and like to contribute with their ideas. In relation to the stakeholder characteristics, it can be mentioned that the issue of writing understandable requirements (Section 4.3), and understanding the stated requirements suggestions, is more complex when the stakeholders are diverse and express their needs vaguely, which is often the case in the market-driven situation. Using natural language, one word may have different meaning to different people. In addition, there is no direct link between the stakeholders and the developers, typically requiring the marketing department to elicit and document requirements. Several of the interviewees requested more communication between the marketing department and developers (Section 4.2). This is especially important in market-driven organisations since the marketing department acts as an interface to the customers and end-users. Although this important interface is the main source of user requirements, some organisations apparently have a more technology-oriented focus where requirements are invented in-house rather than elicited (Section 4.7). Although innovation is necessary in technology-intensive organisations, it may turn focus away from what the users actually need. For example, the focus group participants discussed that users often suggest non-functional requirements, while the functional requirements are in focus internally. Finally, release planning (Section 4.12) is of high importance in the competitive market-driven situation since it is necessary to release products with time-to-market in mind. In order to successfully optimise the user value and development effort for each release, techniques for requirements prioritisation and effort estimation are needed.

The challenges that are special to market-driven organisations are evidently in need of more investigation. However, it is not necessarily those challenges that cause the most problems for the market-driven organisations. The semi-structured type of interviews that were held did not reveal if certain challenges are more important or more acute than others. However, during analysis the *code frequency*, i.e., the number of quotations for each code, was examined and it indicates some patterns regarding which challenges that were more discussed than others. The following challenges were much discussed by all participants: Simple techniques for basic needs, Writing understandable requirements, Requirements traceability and interdependencies, and Release planning based on uncertain estimates. Therefore, they may be of common interest to market-driven organisations in general. The same four challenges also received the highest number of quotations in total, confirming that they are of high importance to

Table 2
Summary of discovered challenges

Challenges	Key problem areas	Special for market-driven organisations	High importance areas
4.1. Simple techniques for basic needs	Tailoring of processes, tool integration, complex specification languages, support for customer feedback		*
4.2. Communication gap between marketing staff and developers	Goal alignment, organisational coherence, information flow, notion of "good requirement"	*	
4.3. Writing understandable requirements	Quality of requirements specification, tracing from requirement to origin, capturing rationale	*	*
4.4. Managing the constant flow of new requirements	Trade-off between elaborate elicitation and information screening, efficient database management, feedback on proposals, finding new requirements among bug reports	*	
4.5. Requirements volatility	Trade-off between volatility and stability, early feedback on product		
4.6. Requirements traceability and interdependencies	Dependency impact analysis, requirements bundling structure, NFR impact trade-off, reduction of redundancy		*
4.7. Requirements are invented rather than discovered	Innovation vs. customer needs, balancing different customers' impact on next release	*	
4.8. Implementing and improving RE within the organisation	Continuous improvement, consistent process enactment, change implementation, staff training		
4.9. Resource allocation to RE	Management support, motivating RE resources, resource competition		
4.10. Organisational stability	Reliance on individuals, co-ordination and communication		
4.11. Selecting the right process	Suitable level of detail, basic and minimal RE process to evolve from, combining sequential and iterative development processes		
4.12. Release planning based on uncertain estimates	Managing cost and value drivers, requirements dependencies, requirements prioritisation, effort estimation	*	*

all of the participating organisations. However, they are not necessarily the most important and cost-effective challenges to deal with in practice. The challenges of high importance are also marked in Table 2. Some of the challenges were discussed more frequently by some participants than others. In the analysis, we attempted to relate those challenges to the organisational characteristics of the particular companies. The purpose was to discover a pattern between the deviation of different answers and the characteristics of the organisation, product or process. However, no clear pattern was visible and therefore no conclusions are drawn based on the relation between organisational characteristics and challenging areas.

5. Discussion

This section is divided in three parts. The first one discusses threats to validity in qualitative designs and the measures taken in the presented study to increase validity. The second part presents our experiences from using a qualitative research approach. Finally, the third part relates our results to the findings in literature.

5.1. Threats to validity in qualitative designs

The quality of a qualitative study relies on the quality of the investigator [31]. In order to obtain practice and experience in interviewing, and try out the interview questions, two pilot interviews were conducted early on. Still there is a risk that quotations become out of context when divided into separate coded segments [6]. To deal with this we have used *observer triangulation*, i.e., multiple observers and interviewers [31], so that three different researchers were involved during interviews and analysis. The coded segments could therefore be discussed so that a common understanding was gained. We have also used *data triangulation* [31] shown in the fact that both interviews and focus groups were used as methods for data collection.

We have chosen to divide the validity issues into the three groups described in [31]: *description*, *interpretation* and *theory*. The main threat to providing a valid *description* of what has been seen or heard lies in the inaccuracy or incompleteness of the data. This has been met by audiotaping all interviews and, later on, transcribing them. Furthermore, each interview was carried out by 2–3 researchers who took extensive notes and collected drawings and sketches that were made by the interviewee.

The main threat to providing a valid *interpretation* is that of imposing a framework or meaning on what is happening rather than this emerging from what is learnt during the involvement with the setting. This does not preclude starting with a set of predefined categories, but these categories must be subjected to checking of their appropriateness, with possible modification. It requires that the researchers demonstrate how the interpretation of the end product was reached. In this research, the threat to interpretation was managed by discussing the interviews and how the different researchers interpreted the interviewees' answers. This was accomplished by having multiple interviewers at all occasions and in addition, a fourth researcher read and commented on the transcripts. Furthermore, the transcripts were analysed using the qualitative analysis tool Atlas.ti [35]. In the tool, the codes made it possible to trace the route by which we have come to a certain interpretation or conclusion.

The main threat to *theory validity* is in not considering alternative explanations or understandings of the phenomena under study. This can be countered by actively seeking data which are not consonant with the theory. We have accomplished this by seeking new types of organisations for our study, where differences in size, age and business type occur.

Validity can also be discussed in terms of *reliability*. In fixed design research, reliability is associated with the use of standardized research instruments, such as formal tests and scales. In qualitative or flexible designs, formal reliability testing cannot be used. However, there are common pitfalls in data collection and transcription that need to be avoided. Among other things, *transcription errors* are explained in [11]. Transcription errors are difficult to avoid due to misinterpretation and mishearing. Inaccurate punctuation or mistyped words can change the entire meaning of a sentence and transcripts do not capture intonation, hesitation, and thought pauses. In this study, the transcriber had also been present at the interview and had heard the answers. In most cases the transcriber was also the one in closest contact with the interviewee and the company, and she/he had thereby a reasonable amount of knowledge about the company culture and language.

Another issue regarding validity is the possibility to generalize the results. *Internal generalisability* is concerned with conclusions drawn within the setting studied. This means that the interviewees or observed situations should not be biased by the researcher. This was managed during sampling as we selected interviewees and organisations from different industrial networks and geographical areas. *External generalisability* concerns conclusions drawn beyond the setting. Qualitative studies rarely attempt to generalise beyond the studied setting, as it is more concerned with characterising, explaining and understanding phenomena under study. Statistical generalisation is often not permitted due to the lack of representative sample. The nature of qualitative designs also makes it impossible to replicate since identical circumstances cannot be re-created. However, the development of a theory can help in understanding other cases and situations. The fact that several of the discovered challenges are acknowledged by all of the participants increases the possibility of transferring the results to other situations. All of the reported challenges are acknowledged by more than two thirds of the participants. Thus, despite the diversity in organisational characteristics, the market-driven type of development seems to create some commonality in challenging areas.

5.2. Experiences from using a qualitative research approach

Qualitative research is rarely used in the software engineering discipline. However, software engineering is in many aspects a social activity and is therefore difficult to understand completely based on quantitative methods. Qualitative methods can give in-depth understanding of social or cultural phenomena. However, some issues make qualitative research difficult in the software engineering discipline. First of all, software is involved in a variety of domains since it is present in various kinds of products. Therefore, the differences in organisational characteristics between the companies involved in the study are large. Thus, it was difficult to know when to stop searching for additional companies to represent the market-driven software development area. The software engineering area is also very fluctuating, which complicates the qualitative research. In our study, it was difficult to extract facts about some of the participating companies since staffing changes frequently, project sizes vary, and documentation is weak. It was also difficult to know whether the process descriptions given by interviewees were examples of how they actually work or how they are supposed to work.

We spent quite a long time adjusting the interview instrument before starting the interviews. While this gave us questions that had been well thought through, spending more time on actually trying the questions out on pilot interviews could have saved more time on the whole. The pilot interviews turned out to be useful, both to adjust the questions and to gain interviewing experience. Later on, multiple interviewers participated during interviews and additional questions were posed to complement the interview instrument. This resulted in a more comprehensive understanding because the different researchers had different focus.

After seven interviews, the data consisted of approximately 100 pages of printed transcripts. This was short enough to manage manually by underlining and marking interesting text segments. However, after the additional seven interviews, the material grew very large. Therefore, it was decided to use a data analysis tool [35] to help us be systematic and rigorous in searching for, and retrieving, data. There is no great conceptual advantage over marking the transcripts physically with code words, but in practice the tool can offer many advantages [6]. It was possible to code at two different sites and then merge the data and the codes together to one single unit [35]. Other benefits include the possibility to search for codes, quotations and words, as well as producing and extracting overlapping codes. These advantages would not have been present with coding on paper. However, one problem was discovered when trying to validate the results. As the results in this paper are written in English and the interviews were carried out in Swedish, the traceability was inferior due to translation problems. To avoid this, memos could have been made in the tool regarding the quotations that were used in the results. In that manner, it would have been easier to search for the memos instead of manually searching through all quotations, which was necessary in our case.

One problem, partly resulting from tool-usage, was that we tended to "over code" the material. The codes were too comprehensive so too many quotations could fit into them. And in fear of losing interesting information, sometimes less interesting quotations were marked as well. This resulted in a massive amount of quotations during analysis; for some categories more than 100 quotations were found. A more strict definition for each category, and routines for adding new as well as more delimited categories, could have been helpful to extract the most interesting quotations and less time could have been spent on analysis of the massive material. This was not as problematic during manual coding in the first study, as the amount of quotations was more apparent in the printouts.

Two researchers with different backgrounds cooperated during coding and analysis. It resulted in slightly different findings and gave us the opportunity to double-check and discuss each other's material from two different viewpoints, and thus increase validity. However, as each person had an in-depth knowledge of a subset of the data it was difficult for one person to get a comprehensive view of the whole material. It was also more complex to include new codes or adjust the coding schema.

5.3. Results related to requirements literature findings

Although Curtis et al. [8] performed their survey several years ago, focusing on large systems development, the study presented here indicates that some problems remain. Communication and co-ordination are still corner stones in software development and project success depends heavily on the skills of the staff involved.

The survey by Hall et al. [15] also confirms that organisational problems, for example lack of skilled personnel and high staff turnover, have a larger impact than technical problems when it comes to requirements engineering. This is in agreement with our results with regard to organisational stability; downsizing negatively affects organisational knowledge and competence and makes it hard to survive with an ad hoc process. Another problem identified by [15] is that sometimes the sales staff agrees to deliver unrealistic system features without considering technical and schedule implications. This was also discussed during our focus group as being a challenge.

Lubars et al. [25] express that the authors were seldom understood when asking the interviewees about which process they follow when writing requirements specifications. This does not seem to be the case in our study as all interviewees had good knowledge about the different steps of their development process. Another difference is that in the paper by Lubars et al. some of the market-driven projects did not produce any written specifications, while this was not discovered in any of the companies we interviewed. Possibly the awareness of the process and the practice to

write specifications have been enhanced since the paper by Lubars et al. was published.

The issue of user participation was stressed by El Emam and Madhavji [12]. In one case they expressed that, *"[Developers] try to force their ideas on the users"*. This was also acknowledged in our study, as there is simply no discrete set of users to invite for participation. Instead, there is a tendency in some of the companies towards increased technology focus, as requirements are invented by developers rather than elicited from potential users. Similarly to [15] and [25], several of the problems encountered in [12] are organisational rather than technical.

Chatzoglou [5] discusses a lack of resources, i.e., people involved, time and money, for RE activities. Lack of resources was also discussed in our focus group meeting, explained by a lack of understanding amongst managers. Lack of resources was also stated as a reason for badly formulated requirements. Most of the participants we asked allocate 10–15% of their time for RE. This was found to be enough by Company C and D, while the larger Company A and B find it to be too little.

As a complement to other surveys, the presented study describes a communication gap between marketing and developers, resulting in insufficient effort estimates and requirements quality. The balance between marketing and developers' requirements decisions is also recognized as a dilemma, especially since new requirements arrive constantly. The use of a requirements database rather than a traditional, monolithic requirements specification is also salient, as well as the urge to group requirements into bundles to ease requirements structuring and work partitioning.

6. Conclusions

This paper has presented several challenging issues with regard to market-driven RE, found during 14 interviews and a focus group meeting. We have not aimed at generalisable results, as the qualitative research approach is intended to characterise and to find variation rather than similarity. This has affected the design of the study since the participating companies had diverse characteristics (see appendix, Table B). Most of the discovered challenges are of an organisational and social nature rather than a technical one. Among the organisational challenges we find issues such as organising projects so that co-ordination and communication is enforced, and obtaining enough resources to carry out RE. Other social issues focus on how to make marketing and development communicate regarding requirements, and how to encourage people to change their way of working, especially when implementing and improving RE in the organisation. Among the technical issues we find issues regarding release planning, techniques for requirements prioritisation and effort estimation, as well as requirements traceability and interdependency problems.

The results of the presented interview study are intended to be useful to researchers in identification of new and industrially relevant research areas. The results can also be useful to practitioners who want to learn from empirically observed obstacles in market-driven software development and base their improvement efforts on such knowledge. In conclusion, the presented results of this study enhance previous industrial surveys on requirements engineering by both corroborating previously found empirical evidence and by adding previously unreported observations to the understanding of industrial requirements engineering.

In Section 4.13 we identified four high importance challenges, which were discussed frequently by all participants: simple techniques for basic needs, Writing understandable requirements, Requirements traceability and interdependencies, and Release planning based on uncertain estimates. Either one of these four challenges can be used as a starting point for research hypotheses and research agendas because of their high importance to the investigated companies. However, these four challenges might not necessarily be the most important and cost-effective ones to concentrate on in practice. Other research needs to be conducted to find out which challenging areas that need most attention and support.

The presented study can be investigated further by increasing the sample of participants with e.g., large- and medium-sized companies, companies developing embedded products, and companies using an agile development approach. These types of companies are only represented to a small extent in the current study and could therefore widen the scope if included in a future survey. In addition, it would be possible to combine the conducted survey with a questionnaire, since the knowledge gained in this study has provided a foundation for constructing closed questions. Typical questions could regard e.g., which kind of development process that is used, which RE tool that is used, which kind of specification language that is used, how requirements traceability is handled, to what extent requirements are invented in-house as opposed to elicited from the market, which kind of requirements prioritisation technique that is used, which kind of cost estimation technique that is used, etc. A questionnaire of that kind could be sent to numerous companies with a market-driven focus to get a comprehensive picture of the processes, tools and techniques that are used in market-driven industry today.

Acknowledgement

The authors give our great thanks to all anonymous interviewees and focus group participants for their time and effort; without your dedication this study would never have been completed.

Appendix A

See Tables A and B.

Table A
Summary of the interview instrument

Characterization	
1.1.	Tell us about the company (number of employees, age, business area, etc.)
1.2.	Tell us about the company's product/products (time on the market, typical customer/end-user, size of product projects, etc.)
1.3.	Tell us about your position in the company (role, daily tasks, responsibility, etc.)
Process issues	
2.1.	What is the procedure when developing a product? (kind of process, activities performed, documentation developed, special cases, evaluations performed, etc.)
2.2.	What is a "requirement" to you?
2.3.	In what way are requirements handled? (requirements process, activities, etc.)
2.4.	What challenges do you face when working with requirements? What has been successful regarding requirements engineering?
2.5.	How much resources are spent on requirements engineering? Continually or in the beginning? How much time would be optimal?
2.6.	What is a "good requirement" to you? And to the company? Is the quality of the requirements assessed? How?
2.7.	What kinds of decisions are taken during the development of a product? What kind of support is needed in those decisions?
2.8.	Is it possible to make decisions too late? What can be the effect in that case?
2.9.	How is it decided what to include in the product? How are the requirements prioritised? What is difficult when deciding what to include in the product?
Artefact issues	
3.1.	How are requirements documented? What information and attributes are documented about the requirements?
3.2.	What support and what tools do you use to document your requirements? What pros and cons do these tools have?
3.3.	How many requirements are handled in a typical project? Who suggests the requirements?
3.4.	What kinds of dependencies between the requirements have you come across? Are dependencies documented? Are dependencies actively looked for?
3.5.	How do dependencies affect product development? How is it handled?
3.6.	Do you group the requirements? How are the requirements groups handled during development?

Table B

	Company A	Company B	Company C	Company D
Total number of employees	1200	65	13	15
Age	20	5	5	2
Product in focus	Software development tool	Embedded software with focus on image processing. 3 product lines	Software development support tool focus on requirements management	Communication tool for distributed groups working with software development
Customer	Software developing companies, mostly in the telecom industry	Consumer electronics retailers and distributors	Software developing companies	Safety critical software developing companies
End user	Developers at software developing companies	Bankers, students, office personnel	Developers and managers at software developing companies	Managers and developers at safety critical software developing companies
Interviewee(s)[a]	1. Req engineer/project manager (1) 2. Product manager (2)	1. Project manager (1) 2. Product manager (1)	1. Founder and managing director (1) 2. Head of developers (1)	1. Project manager (1)
Role/responsibility of the interviewee(s)	1. Co-ordinate resources and clarify, define and prioritise requirements 2. Coordinate between marketing and development, elicit reqs from the market and decide product evolution	1. Introduce process and break down the requirements specification in several parallel projects. 2. Coordinate between marketing, development and production. Suggest and write requirements	1. Product management, release planning, marketing, and sales 2. Regular development tasks, allocate work between developers, and suggest, document and time-estimate requirements	1. Process responsibility and business analysis, involved in daily development tasks
Release size and frequency	New release every 6 months to include new requirements and correct detected errors. Involves 30–60 part-time employees in the development of the product in focus	Includes 25–30 employees for a new product line and 5–15 employees for a new version of existing product. Initiated because of new requirements or when many errors are detected	New release approximately every 6 months. Each release divided into functions, which involves 1–3 employees for 1–10 weeks	The product in focus involves 7 employees at 2 locations. At least one release every month

Table B (*continued*)

	Company A	Company B	Company C	Company D
Process	Defined but not documented, followed thanks to the experienced staff	Introducing process. The ad hoc process works thanks to the experienced staff	Elaborate. Documented and integrated in the tool they develop and use. Based on requirements status. Incremental development within each release	Elaborate. Extreme Programming, pair programming excluded. Incremental development within each release
Requirements documentation	Natural language. Support system database where customers and developers add requirements. A web site with the ones for the current release written on a high abstraction level	Natural language, state charts and UML in the requirements specification. Changes to the specification are rarely documented	Natural language. Database included in the tool where customers and employees can add requirements	Natural language. Customers' requirements suggestions on virtual story cards, which can be followed through the process
Project or process evolution	Not regularly. Have downsized the process but improved requirements awareness	Process under construction. Have increased overall system and product awareness, e.g., within requirements and testing. Introducing project evaluations	Included in the process. Evaluation occurs after each release and has e.g., adjusted the status range	Weekly meetings to sum up the week's problems. Individual time surveillance
	Company E	Company F	Company G	Company H
Total number of employees	12	460	300–400	26
Age	1	2,5	18	16
Product in focus	Software development and visualization tool for integration of electronic business processes	Switch for supporting mobile telecommunication	Network communication solutions for printers	Software tool to support sales configuration for large products
Customer	Large companies that need improved business processes	Telecommunication operators	Systems administrators at companies that use printers	Companies with large configurable products
End user	Developers of electronic business processes	Users of mobile phone technology	Printer users	Sales personnel
Interviewee(s)[a]	1. Founder and managing director (1) 2. Product manager (2)	1. System manager (2) 2. Process engineer (2)	1. Software developer (2)	1. Customer relations (2)
Role/ responsibility of the interviewee(s)	1. Marketing, customer contacts, and release planning 2. Product and technology responsibility. Head of development, control the overall development process	1. Coordinate between product management and development. Refine the high level reqs into detailed reqs 2. Define and improve the RE process. Process introduction. Support the projects with method/ process related issues	1. Regular development tasks; participate in analysis, design and implement requirements (coding), etc.	1. Customer responsibility for their largest customer. Elicit and analyse requirements
Release size and frequency	No regular releases, puts together a project including all employees when many bugs or requirements have been identified	Regular releases, approximately once a year (earlier every 6th month). Involves over 100 developers, divided into teams	No regular releases, puts together a project on demand. Involves 4–8 persons and last about 6 months	Aims at releasing two new versions per year. Involves 6–7 persons
Process	Not documented, but the experienced staff knows the activities by heart	Elaborate and defined. A RUP process especially adjusted towards their needs. Incremental combined with tollgate-based process	Defined and documented. Have a development manual	Defined with documentation templates
Requirements documentation	Natural language in the requirements specifications and UML at a later stage	Natural language and UML, especially in later stages. Several documents, for various levels of detail e.g., one for use cases	Natural language. Have two documents for requirements on different levels of detail	Natural language together with use cases, and other UML diagrams when needed. All reqs are documented in Excel, and for each selected reqs a functional specification is made
Project or process evolution	Not regularly but improvements ideas are discussed	Regular evaluations. Have staff especially responsible for improving the process (interviewee 2)	Not regularly. The project manager adapts the process to the project-specific needs	Not regularly, as part of the development process, but lots of improvements on the way

[a] The interviewees are marked with a (1) if participating in the first part of the study and a (2) if participating in the second part of the study.

References

[1] K.K. Breitman, J.C.S. do Prado Leite, A. Finkelstein, The world's a stage: a survey on requirements engineering using a real-life case study, Journal of the Brazilian Computer Society. 6 (1999) 13–37.

[2] P. Carlshamre, A Usability Perspective on Requirements Engineering – From Methodology to Product Development, Ph D Dissertation No. 726, Department of Computer and Information Science, Linköping University, Sweden, 2001.

[3] P. Carlshamre, Release planning in market-driven software product development: provoking an understanding, Requirements Engineering Journal. 7 (2002) 139–151.

[4] E. Carmel, S. Becker, A Process Model for Packaged Software Development, IEEE Transactions on Engineering Management. 42 (1995) 50–60.

[5] P.D. Chatzoglou, Factors Affecting Completion of the Requirements Capture Stage of Projects with Different Characteristics, Information and Software Technology. 39 (1997) 627–640.

[6] A. Coffey, P. Atkinson, Making Sense of Qualitative Data, Sage Publications, Thousand Oaks, Calif., USA, 1996.

[7] R.G. Cooper, Winning at new products: accelerating the process from idea to launch, Cambridge, Perseus, 2001.

[8] B. Curtis, H. Krasner, N. Iscoe, A Field Study of the Software Design Process for Large Systems, Communications of the ACM. 31 (1988) 1268–1287.

[9] D.E. Damian, D. Zowghi, RE challenges in multi-site software development organisations, Requirements Engineering Journal. 8 (2003) 149–160.

[10] B. Deifel, A Process Model for Requirements Engineering of CCOTS, Proceedings of Workshop on Requirements Engineering Process, Florence, Italy, 1999.

[11] K.L. Easton, J.F. McComish, R. Greenberg, Avoiding Common Pitfalls in Qualitative Data Collection and Transcription, Qualitative Health Research 10 (2000) 703–708.

[12] K. El Emam, N.H. Madhavji, A Field Study of Requirements Engineering Practices in Information Systems Development, Proceedings of the Second IEEE Int. Symp. on Requirements Engineering, York, UK, 1995, pp. 68–80.

[13] M. Fowler, K. Scott, UML Distilled. A Brief Guide to the Standard Object Modeling Language, Addison-Wesley, Reading, Mass., 2000.

[14] B.G. Glaser, A.L. Strauss, The Discovery of Grounded Theory: Strategies for Qualitative Research, Weidenfeld and Nicolson, London, UK, 1967.

[15] T. Hall, S. Beecham, A. Rainer, Requirements Problems in Twelve Software Companies: An Empirical Analysis, Proceedings of the Conference on Empirical Assessment in Software Engineering, Keele, UK, 2002, pp. 1–17.

[16] S.A. Higgins, M. de Laat, P.M.C. Gieles, E.M. Geurts, Managing Requirements for Medical IT Products, IEEE Software. 20 (2003) 26–33.

[17] M.C. Hoepfl, Choosing Qualitative Research: A Primer for Technology Education Researchers, Journal of Technology Education. 9 (1997).

[18] H.F. Hofmann, F. Lehner, Requirements Engineering as a Success Factor in Software Projects, IEEE Software. 18 (2001) 58–66.

[19] E. Honour, Principles of Commercial Systems Engineering, Proceedings from Fifth Annual International Symposium of the National Council on Systems Engineering, St Louis, MO, USA, 1995.

[20] I. Jacobson, G. Booch, J. Rumbaugh, The Unified Software Development Process, Addison-Wesley, Reading, Mass., 1999.

[21] E. Kamsties, K. Hörmann, M. Schlich, Requirements Engineering in Small and Medium Enterprises: State-of-the-Practice, Problems, Solutions and Technology Transfer, Proceedings of the Conference on European Industrial Requirements Engineering, Hammersmith, UK, 1998.

[22] J. Karlsson, Software Requirements Prioritizing, Proceedings of the International Conference of Requirements Engineering, Colorado Springs, Colorado, USA, 1996, pp. 110–116.

[23] J. Karlsson, A Systematic Approach for Prioritizing Software Requirements. Doctorial Dissertation, Department of Computer and Information Science, Linköping University, 1998.

[24] L. Karlsson, Å.G. Dahlstedt, J. Natt och Dag, B. Regnell, A. Persson, Challenges in Market-Driven Requirements Engineering - an Industrial Interview Study, Proceedings of the Eighth Int. Workshop on Requirements Engineering: Foundations of Software Quality, Essen, Germany, 2002, pp. 37–49.

[25] M. Lubars, C. Potts, C. Richter, A Review of the State of the Practice in Requirements Modelling, Proceedings of the First IEEE Int. Symposium on Requirements Engineering, San Diego, California, USA, 1993, pp. 2–14.

[26] M.D. Myers, Qualitative Research in Information Systems, MIS Quarterly, MISQ Discovery. 21 (1997) 241–242.

[27] U. Nikula, J. Sajaniemi, H. Kälviäinen, A State-of-the-Practice Survey on Requirements Engineering in Small- and Medium-Sized Enterprises, TBRC Research Report 1, Telecom Business Research Center Lappeenranta, Lappeenranta University of Technology, 2000.

[28] R. Novorita, G. Grube, Benefits of Structured Requirements Methods for Market-Based Enterprises, Proceedings of the International Council on Systems Engineering (INCOSE) Sixth Annual International Symposium on Systems Engineering: Practices and Tools, Boston, Mass., USA, 1996.

[29] M.Q. Patton, Qualitative Research and Evaluation Methods, Sage Publications, Calif., USA, 2002.

[30] C. Potts, Invented Requirements and Imagined Customers: Requirements Engineering for Off-the-Shelf Software, Proceedings of the Second IEEE International Symposium on Requirements Engineering, IEEE Computer Society Press, York, UK, 1995, pp. 128–130.

[31] C. Robson, Real World Research, Blackwell, Oxford, 2002.

[32] P. Sawyer, Packaged Software: Challenges for RE, Proceedings of the Sixth Int. Workshop on Requirements Engineering: Foundations of Software Quality, Stockholm, Sweden, 2000, pp. 137–142.

[33] P. Sawyer, I. Sommerville, G. Kotonya, Improving Market-Driven RE Processes. Proceedings of the Int. Conf. on Product Focuesed Software Process Improvement, Oulu, Finland, 1999, pp. 222–236.

[34] P. Sawyer, Packaged software: implications of the differences from custom approaches to software development, European Journal of Information Systems. 9 (2000) 47–58.

[35] www.atlasti.de (last visited February 2006).