

Needs and Challenges for a Platform to Support Large-scale Requirements Engineering

a Multiple-case Study

DAVIDE FUCCI, HITeC, University of Hamburg

CRISTINA PALOMARES, DOLORS COSTAL, XAVIER FRANCH, Universitat Politècnica de Catalunya

MIKKO RAATIKAINEN, University of Helsinki

MARTIN STETTINGER, IST, Graz University of Technology

ZIJAD KURTANOVIC, University of Hamburg

TERO KOJO, LARS KOENIG, The Qt Company

ANDREAS FALKNER, GOTTFRIED SCHENNER, Siemens

FABRIZIO BRASCA, WindTre Italia

TOMI MÄNNISTÖ, University of Helsinki

ALEXANDER FELFERNIG, IST, Graz University of Technology

WALID MAALEJ, University of Hamburg

Background: Requirement engineering is often considered a critical activity in system development projects. The increasing complexity of software as well as number and heterogeneity of stakeholders motivate the development of methods and tools for improving large-scale requirement engineering. *Aims:* The empirical study presented in this paper aim to identify and understand the characteristics and challenges of a platform, as desired by experts, to support requirement engineering for individual stakeholders, based on the current pain-points of their organizations when dealing with a large number requirements. *Method:* We conducted a multiple case study with three companies in different domains. We collected data through ten semi-structured interviews with experts from these companies. *Results:* The main pain-point for stakeholders is handling the vast amount of data from different sources. The foreseen platform should leverage such data to manage changes in requirements according to customers' and users' preferences. It should also offer stakeholders an estimation of how long a requirements engineering task will take to complete, along with an easier requirements dependency identification and requirements reuse strategy. *Conclusions:* The findings provide empirical evidence about how practitioners wish to improve their requirement engineering processes and tools. The insights are a starting point for in-depth investigations into the problems and solutions presented. Practitioners can use the results to improve existing or design new practices and tools.

CCS Concepts: • **Software and its engineering** → **Requirements analysis**;

Additional Key Words and Phrases: large-scale requirement engineering, user participation, stakeholders productivity, recommender systems, case study

Authors' addresses: Davide Fucci, HITeC, University of Hamburg, Hamburg, Germany, fucci@informatik.uni-hamburg.de; Cristina Palomares, Dolors Costal, Xavier Franch, Universitat Politècnica de Catalunya, Barcelona, Spain, cpalomares,franch,costal@essi.upc.edu; Mikko Raatikainen, University of Helsinki, Helsinki, Finland, mikko.raatikainen@helsinki.fi; Martin Stettinger, IST, Graz University of Technology, Graz, Austria, martin.stettinger@ist.tugraz.at; Zijad Kurtanovic, University of Hamburg, Hamburg, Germany, kurtanovic@informatik.uni-hamburg.de; Tero Kojo, Lars Koenig, The Qt Company, Espoo, Finland, tero.kojo,lars.koenig@qt.io; Andreas Falkner, Gottfried Schenner, Siemens, Vienna, Austria, andreas.a.falkner,gottfried.schenner@siemens.com; Fabrizio Brasca, WindTre Italia, Milan, Italy, fabriziogabrio.brasca@windtre.it; Tomi Männistö, University of Helsinki, Helsinki, Finland, tomi.mannisto@cs.helsinki.fi; Alexander Felfernig, IST, Graz University of Technology, Graz, Austria, alexander.felfernig@ist.tugraz.at; Walid Maalej, University of Hamburg, Hamburg, 20257, maalej@informatik.uni-hamburg.de.

Manuscript submitted to ACM

1

1 INTRODUCTION

Requirements are often considered the basis for all subsequent development, deployment, and maintenance activities. Poorly implemented Requirements Engineering (RE) presents significant risks for a project [3], including its cancellation or additional costs [13]. Gartner research found that requirements are the third source of product defects and the first source of delivered defects for service projects. Accordingly, the cost of fixing defects ranges from \$70 at the requirements phase to \$14,000 in production phase¹.

Nevertheless, RE often receives little project effort [9] and, in spite of the advances in the field, practitioners still struggle with it [10, 19]. For instance, a recent study [7] reports that the requirements definition is a challenge for practitioners.

At the same time, the latest advancements in machine learning and natural language processing bear new potentials to support decision-makers in the context of RE [15, 17]. As organizations are constantly looking for ways to produce novel products and services, a platform—embedding the above technologies to support the RE process—can increase stakeholders’ and customers’ satisfaction, improve time to market, and reduce costs.

The research project OPENREQ² aims to develop such a platform for companies dealing with large-scale requirements—e.g., in the order of thousands per project [25]. In particular, OPENREQ covers industrial scenarios related to bid management for railways, community-driven cross-platform development tools, and telecommunication services. At the start of the project, it was necessary to clarify the industry needs regarding the lack of support for RE tasks. Moreover, the close partnership with industry presented a chance to reduce the divide between practitioners and researchers [7]. In fact, as requirements are usually project-specific and business critical, it is challenging for practitioners to share insights regarding their processes and artifacts. Accordingly, for researchers it is difficult to form a clear understanding of RE in practice [8]. Understanding the state-of-practice is the first step to support the development of the OPENREQ platform in a problem-driven fashion. As recent studies [6, 7] report that stakeholders are overwhelmed by the number of decisions to make when dealing with requirements, we focus our investigation and initial project efforts on the area of *individual stakeholder support* [21].

This paper grounds in empirical evidence the desired characteristics for a platform supporting large-scale RE in industrial context. We base our findings on the pain-points associated with how RE tasks are currently approached by three companies in different domains. Moreover, we aim at identifying challenges that the companies anticipate when using a platform such as OPENREQ. To that end, we conducted an exploratory, multiple case study [35] within the OPENREQ partner company units responsible for RE.³

The paper makes two contributions. First, it offers in-depth insights into the state-of-practice of very-large RE and defines the needs for individual stakeholders. Second, it generates new insights useful to design a platform supporting such needs, and the potential challenges to be taken into account when adopting it.

Paper organization. Section 2 surveys the state-of-practice related to current problems and needs in RE. Section 3 introduces the exploratory case study including research questions, method, and data. We present the results and summarize the main findings in Section 4. We discuss the implication as well as the limitation in Section 5. Finally, Section 6 concludes the paper.

¹[https://www.gartner.com/doc/1753116/hype-cycle-application-development-](https://www.gartner.com/doc/1753116/hype-cycle-application-development)

²https://cordis.europa.eu/project/rcn/206364_en.html

³Due to confidentiality agreement the full data (e.g., interviews transcript) cannot be disclosed.

2 RELATED WORK

Currently, there seems to be a chasm between the state-of-the-art research and practical RE work [7]. Although there are several empirical studies dealing with the challenges faced by industry during the RE stage, only a small part investigates what industry expects from RE research—e.g., concerning tools and approaches.

Several studies show that current tool support for RE is not enough. Carrillo de Gea *et al.* [4] presents the results of a survey administered to 38 RE tool vendors with the purpose of gaining insights into tools capabilities and the extent to which tools support RE processes. Results show that RE tools need improvement, principally regarding requirements modelling, open data model (i.e., available data at any time from an external tool) and data integration features; in particular, the integration of data from external sources. A survey of 307 practitioners by Maalej *et al.* [16] shows that tool support for complex RE tasks is often poor or absent, with requirements negotiation and planning being the least supported. In contrast, this study takes the perspective of the tool final users, suggesting what are features relevant for them given the problems they are experiencing in their domain.

Requirements traceability is the main focus of the semi-structured interview study conducted by Rempel *et al.* [26]. They conducted 20 interviews in 17 companies to investigate how requirements traceability is done by practitioners and the challenges that they face. Findings suggest that a traceability strategy defined up front is indeed required because it is a complex task to determine all suitable trace paths for a project. The decision for or against trace paths between requirements and artifacts requires a detailed understanding of the project engineering process and goals.

Palomares *et al.* [22] conducted an exploratory survey to investigate the state-of-practice in the reuse of requirements. The survey was based on an Internet questionnaire with 71 responses from requirements engineers with industrial experience. Although they found that the majority of respondents declared some level of reuse in their projects, only a minority of them declared such reuse as a regular practice. Ignorance of reuse techniques and processes is the main reason preventing wider adoption, followed by needing a high initial investment, and the complexity of implementing requirements reuse.

Raatikainen *et al.* [23] describes the state-of-practice of RE in the nuclear energy domain in Finland on the basis of a descriptive case study focusing on safety automation systems of nuclear power plants. Data was collected by interviewing two domain experts representing public authority and five experts working at three power companies. The identified challenges of RE are in aging, knowledge transfer, traceability, communication, and tool support. As for RE tools, results highlight the need of a tool that supports relationships, hierarchies and traceability, and the difficulty to incorporate them into an existing environment.

The work of Wnuk *et al.* [34] presents the results of 13 interviews in three companies with the objective of understanding the challenges in scoping large RE projects and how these challenges are addressed by the studied companies. The identified challenges are related to information overload (e.g., difficult to make relevant grouping, hard to have a global overview of the system, how to get deep knowledge, how to acquire the knowledge to make a decision, information cannot be trusted), requirements reuse (inability to reuse requirements), and requirements quality (unclear and complicated requirements).

Sethia *et al.* [29] used an online survey to establish and validate an empirical model about the behavior between requirements elicitation issues and project performance. The study takes into account the responses of 203 participants involved in RE from different companies and domains. The results identify different challenges related to the elicitation of requirements. First, there are challenges related to the diversity of stakeholders—e.g., large differences in users' needs, effort needed for reconciling requirements from various users. Second, there are challenges related to communication

(e.g., users had significant problems when communicating requirements). Third, there are problems requirements volatility (i.e., related to the extent of changes that the requirements undergo during the project life cycle). Finally, there are problems related to requirements quality, specially concerning requirements that are ambiguous and unstable. This study confirms the issues identified in [22, 23, 26, 29, 34] while elaborating on how to address them in three different domains.

One of the goals of Hiisilä *et al.* [11] is to investigate what are the challenges in the RE process of a customer organization in an outsourced development environment. With that purpose, they conduct a case study in a Finnish insurance company performing 17 interviews and analyzing 15 large projects. In addition, they conducted five workshops to validate the results. The case study identifies some challenges related to the elicitation and specification of requirements. Some of them are related to information overload (e.g., scoping and planning the project, and prioritization of requirements), some to stakeholders (e.g., large number of stakeholders, lack of cooperation in the early RE stages, management involvement as a stakeholder), some to traceability (from the business requirements to the benefits received), and some to requirements quality (arriving to a common understanding and reaching comprehensive requirements for all parts involved in the RE process).

To identify key industry needs, Sikora *et al.* [30] conducted an in-depth study with representatives from large, internationally-operating companies in the domain of embedded systems in Germany. They interviewed ten practitioners with a clear view of the RE needs of their companies. The authors show that support is needed for traceability, testability, completeness, and verifiability.

On top of recognizing the same problems reported in [11, 30], this paper presents the needs that experts, working in companies in three different domains and countries, perceive and the challenges anticipated when addressing them.

Méndez *et al.* present NAPIRE, a globally distributed family of surveys to study the state-of-the-practice in RE, as well as the results of the first run of the survey in Germany [6] and a second run conducted all over the world [7]. Different challenges related to RE are identified, covering aspects such as missing dependencies, insufficient stakeholder involvement, flaws when communicating with customers, different requirement quality challenges, and the lack of time dedicated to RE activities.

Among the reviewed studies, Méndez *et al.* [7] is the only one covering the RE needs of companies in several domains and geographical areas. We complement this study by providing qualitative evidence supporting the existence of challenges while focusing on companies dealing with large-scale requirements.

3 CASE STUDY DESIGN

The objective of this study is to *understand what companies consider an innovative RE platform*. Although the interviews covered a broader spectrum of aspects, in this paper, we focus on techniques that support individual stakeholders, meaning techniques that help them, as individuals, during the RE process. These have to be considered in contrast to the techniques that consider groups of stakeholders in RE, which takes into account the preferences and needs of all the stakeholders in the group. We consider this a multiple (i.e., performed through several cases), holistic (i.e., a single unit of analysis is studied in each case [35]), and exploratory (i.e., the goal is to seek new insights rather than test existing assumptions) case study, reported according to the guidelines of Runeson *et al.* [27]. Data were directly collected using semi-structured interviews with ten company representatives and analyzed using open, axial, and selective coding. The research process is reported in Figure 1.

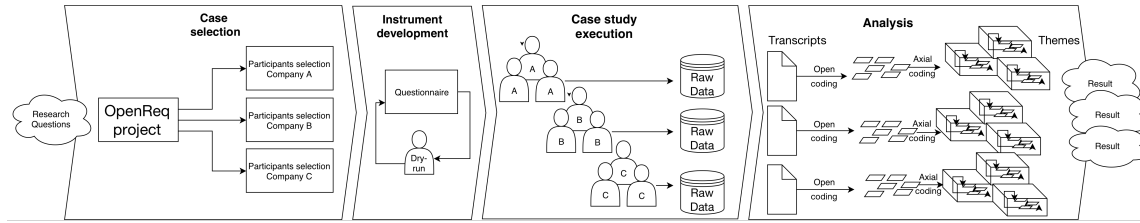


Fig. 1. Research phases for the presented case study.

3.1 Research Questions

Our case study was driven by the following research questions:

- *RQ1*: What are the *pain-points* faced by the companies in the way they currently deal with large-scale RE?
- *RQ2*: What are the *needs* that a platform supporting large-scale RE should address?
- *RQ3*: What are the *challenges* of introducing such a platform in the case companies?

RQ1 is necessary to get an understanding of the companies status quo and their current problems. Thereafter, we investigate the main objective of this research—what needs should the platform support—in RQ2. Finally, we formulate RQ3 to understand what are the challenges that the companies foresee with the introduction of approaches addressing their needs.

3.2 Case and Subject Selection

All the three companies deal with a large number of requirements; however, several other factors vary between them as summarized in Table 1. Table 2 provides details about the interview participants.

3.2.1 The Qt Company. (henceforth, Company A). The company develops tools and solutions for cross-platform software development with a focus on embedded devices. The company has several sites worldwide and consists of 200 employees. The company is actively involved with open-source development as part of its business is in maintaining and further developing tools and framework freely and openly available to the community. Such solutions are then tailored to the needs of the company clients. The company has a robust Agile development process in which requirements originate from the community as well from the customers.

The interviewees from Qt are three, one product manager and two senior R&D managers. The representative of the OPENREQ project in Qt recruited them due to their active involvement in maintaining and improving the RE process within the company. The participants worked in the Qt R&D site in Germany.

3.2.2 Siemens Mobility AG. (henceforth, Company B). The company, part of a larger conglomerate, operates as a supplier and system integrator in the railway sector. The company operates worldwide and has approximately 28,000 employees. Managing bids and requests for proposals (RFPs) for safety systems issued by national providers and ensuring that offers comply with technical specifications, are important tasks of the work process. One R&D manager (i.e., the OpenReq representative), one senior R&D, and two bid managers were interviewed during the study. The participants work at the Austrian location of Siemens.

3.2.3 WindTre Italia SPA. (henceforth, Company C). The company is one of the largest telecom operators in Italy and part of a conglomerate which provides services worldwide. As it is often the case for companies operating in this

Table 1. Overview of the case companies.

	Company A	Company B	Company C
Type of company	Software product	Infrastructure development	Infrastructure development
OPENREQ scenario	Cross-platform development tools	Bid management for railways	Telecommunication services
#employees in RE	100 in R&D, 10 in PM, no specific RE	> 200	no specific in RE
#employees in typical project	All	20-30 in different roles	> 100
Distributed	Yes	No	No
Source of requirements	Bespoke, community (market-driven)	Request for proposal (bespoke)	Market driven
Main RE tool	Atlassian JIRA issue tracker	IBM Rational DOORS, Microsoft WORD	REMEDY, Microsoft WORD, Microsoft EXCEL
Process model	Iterative	Waterfall-like process for the bid phase and iterative or Agile approach for the development phase depending on the specific project	Currently being reworked towards Agile
Release duration for a typical project	Major releases every six months, minor releases even weekly	For development projects, preparation lasts several weeks to several months	Several months
#requirements in a typical project	several thousands	several thousands	hundreds to thousands

sector, their requirements are market driven—i.e., they produce solutions sold to customers on an open-market [24]. The unit covered in this study consists of 7,000 employees and has a strong focus on RE activities, while the majority of development is outsourced to third parties.

The interviewees from WindTre are two senior R&D managers (one of whom is the company representative for OPENREQ) and a senior network engineer manager. The project representative recruited the rest of the interviewees. They worked in two of the company sites in Italy.

Table 2. Interviewees. ID indicates the company.

ID	Role	Role experience	Company experience
A1	Product manager	7 years	9 months
A2	Senior R&D	1.5 years	13 years
A3	Senior R&D	1.5 years	10 years
B1	R&D manager	10 years	25 years
B2	Senior R&D	20 years	20 years
B3	Senior bid manager	10 years	20 years
B4	Senior bid manager	12 years	20 years
C1	Senior R&D	8 years	16 years
C2	Senior R&D	20 years	22 years
C3	Senior engineer	18 years	18 years

Table 3. Summary of the questionnaire used during the interviews.

Phase	Topics	# questions	Duration (in minutes)	Research question
1	- Background of the company and participants - Status-quo regarding processes and tools for RE - Problems encountered in the current status	45	120	RQ1
2	- Ways to improve the current status and addressing pain-points	7	50	RQ2
2b	- Improvements specific for individual stakeholders support	8	40	RQ2
3	- Challenges in the implementation of Phase 2	5	20	RQ3
3b	- Challenges in the implementation of Phase 2b	2	10	RQ3

3.3 Data Collection Procedures

We carried out semi-structured interviews with the participants between March and May 2017. For each company, the interviews were co-located and took place at the company premises with all the participants at the same time. Each interview session lasted for approximately four hours. We took 15 minutes breaks every 45 minutes. The interviews were carried out in English, and the audio was digitally recorded upon verbal consent from the participants.

The instrument for data collection was a funneled questionnaire [28]—posing questions from general, based on the research questions, to specific ones. Therefore, we used open-ended questions to let the participants reflect on improvements, needs, and challenges for their RE process and tools; and closed questions to explicitly obtain information about specific topics (e.g., status-quo, attitudes towards specific solutions). After a dry-run, we internally reviewed the questionnaire and adjusted it to fit into the allotted time. When doing so, we allowed room for other questions—not included in the script—of potential interest towards our objectives.

A summary of the questionnaire is reported in Table 3, whereas the final interview script is available at the authors' website ⁴. Before the first phase, we presented the objectives of the interview and the case study and explained how the data from the interview will be handled and used.

Up to four interviewees were present during each session; the questions we asked everyone all questions when possible. Their answers, comments about others' answers, and conversations were recorded. When an interviewee in a higher position was present (i.e., product manager), we preferred to hear first the answers from interviewees in other positions (i.e., engineers) to limit biases. After the interview, we used a professional service to obtain a timestamped, word-by-word transcript of the audio recording.

3.4 Analysis Procedures

We performed qualitative data analysis based on the transcripts obtained as described in [20]. We coded the transcripts using a line-by-line approach, in a semi-exploratory fashion. The initial codes were based on the main pain-points in RE activities, participants wish for solutions, and foreseen challenges. For each transcript relative to a company, each relevant statement in the transcribed interview was assigned codes by two researchers following an open coding approach [2].

These codes were subsequently grouped to form themes (e.g., tools, roles, and channels from which requirements arise) and their relationships were captured by applying axial coding [2]. The data was analyzed using Atlas.ti⁵, which allows sorting the data according to codes and themes (e.g., by roles, problem, solution) for easy identification of patterns. We did not follow a continuous comparison analysis across cases [28]. We coded the three cases independently,

⁴<http://www.upc.edu/gessi/OpenReq/InterviewGuide.pdf>

⁵<https://atlasti.com>

with a researcher in charge for a case, and one or two others supporting her with the review of the codes. Therefore, common and contrasting themes were later identified for each single case. This decision was taken because, in the context of the OPENREQ project, the solutions will be evaluated in the companies specific contexts. Therefore, a constant comparison between each case (either during the interviews or coding) could have jeopardized this aspect.⁶ After the individual case coding, we compared and reconciled the concepts reported in Section 4.

Finally, we applied selective coding to choose the themes relating to individual stakeholders support [2].

3.5 Validity Procedures

The questionnaire instrument was reviewed by the researchers in two iterations, followed by a complete dry-run to assess possible inconsistency or replicated questions. The transcript was performed by a third party to avoid introducing researcher bias in the raw data. At least two researchers reviewed the transcripts to check its correctness, and gaps were then filled-in by interviewees.

Triangulation of the analysis was achieved by having different groups of two to three researchers checking the open codes in parallel. We maintained traceability between the transcript and the coding, and between the coding and the reporting by using a state-of-the-art software solution for qualitative data analysis which streamlines the process.

Although the case study lasted for a limited amount of time, the researchers started collaborating with the companies already few months before the case study was planned; some of the authors have long-term cooperations with Company A and Company B.

4 RESULTS

This section presents our findings, and answers the research questions. The final set of codes, reported in Table 4, includes the five pain-points the companies felt the most regarding their current support to individual stakeholders in RE, the ten needs that should be addressed, and the 12 challenges that implementing such needs poses. In the following, wherever there is a quote from the interviews, there is also between brackets the id of the interviewee who voiced the quote.

4.1 RQ1: Pain-points faced in large-scale RE

We report the pain-points which occurred the most, and were shared by at least two of the three case companies.

P1. Information overload. All the companies have to deal with difficulties due to the size and range of information needed to carry out RE activities. Stakeholders have to look for relevant pieces of knowledge necessary to make decisions in large, fragmented repositories of data. Parsing such information—e.g., for requirements maintenance tasks—is time-consuming and perceived as disheartening: *“There is a certain level of information overload, especially with the mailing list. So I used to read everything, not anymore. I just can’t read that anymore, it’s just too much under discussion.”* [A2]. For example, acquiring data to support requirements elicitation is not a problem, but its analysis is, resulting in diminished productivity: *“Every project is a bit different but we have all our documents in Sharepoint, and it is hard to find what we need. The problem is that looking at these documents, just to find a single interesting sentence, takes a long time.”* [B2].

P2. Tools limitations. A second pain-point perceived by the three companies is the limitation imposed by existing tools.

⁶For the same reason, we do not provide an explicit comparison among the cases and do not explicitly map the codes into RE activities due to their different processes.

Table 4. Themes (pain-points, needs, and challenges) as perceived by the companies.

Pain-point (RQ1)	Needs (RQ2)			Challenge (RQ3)							
	A	B	C	A	B	C					
P1. Information overload	X	X	X	N1. Data analytics	X	X	X	C1. Data sources	X		
				N2. Visualization	X		X	C2. Heavier process	X		X
				N3. Effort estimation	X	X		C3. Lack of metrics	X	X	
P2. Tools limitations	X	X	X	N4. Integration with upstream tools	X		X	C4. Changes to process		X	X
				N5. Trace decision-making	X	X		C5. Community values	X		
				N6. Stakeholder feedback	X	X	X				
P3. Dependency between requirements	X	X		N7. Configurable dependencies	X	X		C6. Indirect dependencies	X	X	
				N8. Dependency at different levels	X	X		C7. Dependency interpretation	X	X	
P4. Requirements reuse		X	X		N9. Reuse from other domains		X	X			X
											X
P5. Stakeholder identification	X	X		N10. Personal recommendation	X	X		C10. Stakeholder involvement	X	X	
							C11. Accuracy of the result		X	X	
							C12. Privacy of the data	X	X	X	

On one hand, Company A main RE tool is too focused on low-level requirements (e.g., bugs to be fixed) and can discourage their community to request larger features or improvements of the products. *“Jira is really driven by bugs. It’s lacking when it comes to feature requirements, so it is awkward for people to file more high-level stuff.” [A2].* On the other hand, Company B uses a tool that does not offer the possibility to add detailed information to document decisions. Thus, Company B had to come up with a workaround: *“We fill in the information in DOORS, but then we use hashtags to insert comments, so we can filter it later.” [B4].* However, the interviewees felt that relying on this approach, rather than on tool support, can hinder the traceability between the initial request for proposal and the final requirements.

P3. Dependency between requirements. The stakeholders involved in the companies RE process struggle to identify requirements relationships which can support tasks such as prioritization.

The similarity between requirements is a type of dependency of particular importance for Company B. In their case, similar requirements are obtained not only from previous projects but also from other sources, such as stakeholders who are experts in a particular infrastructure: *“There might be similar requirements because the projects are similar. But it can also be that requirements are similar because they are coming from the same stakeholder.” [B1].*

Company A leverages dependencies between requirements for prioritization tasks. However, understanding the value of managing such dependencies in the long term is a problem: *“Relationships between requirements gives us sometimes a kind of guidance for prioritization. For instance, when addressing an issue is required to fulfill another. This dependency raises the question of whether something with no or small benefit should have higher priority than others.” [A1].*

P4. Requirements reuse. The problem of identifying dependencies (P3), affects the reusability of requirements. Company A has basic strategies to reuse requirements from the same sub-domain.

In Company B, requirements are not reused, but previous knowledge about the project helps the requirements manager to assign a new requirement to a stakeholder. However, this process is not automated and relies on expert knowledge. *“Requirements are not directly reused because they are new and not written by the requirements manager. If there’s a requirement that is similar to another bid project, maybe he will remember and look up the person responsible for it.”* [B4].

P5. Stakeholders identification. The companies follow different strategies to assign the appropriate stakeholder for an RE task. The company culture has a role in the realization of such strategies.

For example, in Company A, deciding who should be assigned to a requirement is a process driven by open-source values, such as open communication: *“Overrule the initial decision about who should work on something is not a big deal as long as it is discussed together.”* [A2].

In the other cases, the decision is driven by the internal company structure and the responsibilities of each department: *“Usually, according to the organization of the company, we know what department is involved in a particular process, so we usually invite these people to start the process [of assigning requirements].”* [B1].

Accordingly, the companies perceive that a fixed correspondence between domain and (expert) stakeholder poses a risk related to employees turnover (i.e., low bus factor [18]).

Moreover, both companies reported a “domino effect”—a stakeholder who does not believe to be a good fit to process a requirement assigns it to someone else causing delays.

4.2 RQ2: Needs to be addressed by the platform

In this section, we report the needs the companies reported. The needs are mapped onto the pain-points presented in Section 4.1

N1. Data analytics. The three companies agreed that leveraging data analytics is necessary to support individual stakeholders in reducing information overload. In particular, Company A suggested analytics as a way to generate a roadmap to reduce the time stakeholders spend for planning: *“The use of analytics could get me an understanding of the relative importance of the tasks I need to do—what should I really do next.”* [A3]. Company B offered a specific wish for a feature—filtering out information (i.e., from the text) that does not constitute a requirement: *“A good starting point for us would be differentiating between what is a requirement and what is not.”* [B1].

N2. Visualization. To grasp the large amount of data available to them, Company A and Company C identified the need of a visualization—e.g., a dashboard. In particular, they expressed their need to not only graphically present requirements meta-data (e.g., status, implementation time) but also the decision-making process. To that end, Company A suggested the use of a heatmap to understand in which phase the stakeholders spend more time.

N3. Effort estimation. The companies need a way to measure the effort necessary to complete a task (e.g., regarding of time) as a way to better plan and limit individual stakeholders from being overloaded. For Company B, this metric is needed to make deadlines and the associated effort more explicit. *“What we are looking for is to have some indicators, such as «if you continue at this rate you will not finish in time» or something similar.”* [B2]. At the moment, none of the companies use a data-driven effort estimation approach (e.g., [31]) but rely on expert evaluation.

N4. Integration with upstream tools. Company A and Company C perceive as beneficial the integration of their current tools with the ones used to support upstream activities, such as customer satisfaction and sales. For example, Company A identified the integration with tools, such as Salesforce, as necessary to get closer to the customers and better identify their needs. *“We would like to see more direct feeding of information from a meeting that a salesperson has with a customer. I know that they have to report how that meeting went, but I don’t see it.”* [A2]. Company B does not have such need as the requirements originate from, for example, governments with whom interaction is infrequent.

N5. Trace decision-making process. It is essential for Company A and Company B that tools facilitate the traceability of their decision. In Company B such possibility is missing.

N6. Stakeholder feedback. The possibility to give feedback to stakeholders is deemed as a pivotal need by the three companies. However, such feature is not present in the tools currently in use. The companies not only expressed the need for explaining the results to the stakeholder but also to incorporate the stakeholder feedback as part of the results.

N7. Configurable dependencies. Although the companies have in place mechanisms to identify dependencies between requirements, they acknowledge that a “one-size-fits-all” definition does not exist. Accordingly, there is a need for a configurable way of expressing dependencies between requirements.

In the case of Company B, the similarity between requirements is seen as a type of dependency (i.e., duplication of some functionalities) which should be defined differently by each stakeholder: *“The term similarity can be seen differently. So, one should configure his measure of similarity because we don’t think that we can come up with a fix set.”* [B3].

In Company A, dependencies should be identified manually, posing a burden on the users—e.g., when filing a request after consulting the issue tracker. For a complex product, this is not feasible, and they recognize the need to automate the process as much as possible.

N8. Dependency at different levels. Dependencies between requirements should be identifiable at different levels—not only within the same requirement document or project but also across projects: *“Viewing similar requirements either in a project or even between projects, but still identifying if two requirements have been talking about the same thing.”* [B1]. In particular, the companies are interested in identifying contradicting requirements between projects.

N9. Reuse from other domains. The only need that Company B and Company C associate with *P4*, is to base their reuse strategies on other, related domains in which the company has expertise.

N10. Personal recommendations. The interviewees expressed the need to use a (personal) recommender system to identify stakeholders and the requirements they should tackle. *“[...] supporting me with the assignment of the stakeholders, somehow implies that a system has some knowledge about previous decisions, learned from past requirements.”* [B1]. For Company A, this need translates into a support system for the triaging of bugs filed in their issue tracker. *“What could certainly help is if the system can suggest me an assignee [for a bug].”* [A1].

4.3 RQ3: Challenges in introducing platform

In this section, we report an overview of what the companies perceive as the main challenges for the implementation of the needs identified in Section 4.2.

C1. Data source. According to Company A, the data for analytics should be carefully chosen otherwise resulting in more information overload: *“The problem is that if you start collecting one [type of data], then there are fifty different other things that you want, and then it becomes questionable very quickly.”* [A3]. The other two companies did not report this as a challenge due to the predictability of the requirements sources.

C2. Heavier process. Company A perceives that a data analytics solution can hinder the “agility” of their process: *“We also have to watch out that that it doesn’t make the whole process longer. For example, if the tool pushes you to look through all this other information, you just give up because you just want to file a bug and be done with it.”* [A2].

C3. Lack of metrics for effort estimation. Realizing a mechanism to estimate the effort required to complete an RE task necessitates specific metrics, not currently collected within the companies.

C4. Changes to the process. The interviewees expressed their worries regarding tool integration, fearing that it will drive a change in their usual process: *“the other difficulty is that, to follow the new process, it is necessary to compose the requirement in another way.”* [B2].

C5. Community values. With JIRA integrated with other tools (e.g., CRM) to document decisions taken in Company A, it is necessary to respect the community values when communicating them. *“I’m all for transparency, but this implies that sometimes you have to argue with customers, and explain them why you are downgrading the priority for the delivery of a component.”* [A2]. Besides, they fear that customers will not accept decisions exclusively based on the tool results.

C6. Indirect dependencies. The companies perceived indirect dependencies as a hurdle to address requirement dependencies in general. This is particularly challenging when requirements are contradicting: *“Identifying contradicting requirements is an hard task. Perhaps one requirement requires a certain approach and another requires another approach, but two those approaches are not compatible.”* [B1].

C7. Dependency interpretation. A challenge, reported by both companies, is the different interpretations that stakeholders attribute to dependencies. Giving the possibility to identify dependencies at different levels is risky as stakeholders have to interpret or assess dependencies from unfamiliar projects.

C8. Employees turnover. Company C indicated the change in personnel dealing with requirements as a threat to requirement reuse: *“Sometime the system where you can reuse a requirement could change. But also the people in charge of the system could change. This is, in my opinion the major difficulty to reuse completely a requirement.”* [B3]. The companies perceive as relevant the human factors in reuse.

C9. Requirements volatility. Company C identified requirements volatility—i.e., becoming less relevant over time—as a problem for reuse, especially when changes are exogenous to the company. *“I think it’s hard for a process, or a solution... everything changes very fast. For example, for new regulation, we changed our approach many times”* [C1].

C10. Stakeholder involvement. Identifying stakeholders necessitates their modeling (e.g., based on preferences, career profile, attitudes). Company A and Company B perceived finding and leveraging expert knowledge as a barrier: *“Even if we go through all the old projects, we need an expert... That’s the hard part.”* [B3].

C11. Accuracy of the results. Two of the companies observed that it is not only essential to have a system that helps them during RE, but also that the results for complex tasks (such as recommendations or data analytics) should be as accurate as possible. Otherwise, the effort needed for correcting the results will discourage users from using the system. *“For me, this [a tool] is successful if it can analyze a lot of data and give us the same results that a human can give.”* [B2]. In other words, as proposed by Berry [1], the tool should achieve better recall than a human working on the task manually.

C12. Privacy of the data. All companies agreed that it might be difficult to have access to the data that a RE system to support individual stakeholders might need (e.g., information about users or monitoring the use of the tool). This is due to company privacy policies, or due to legal aspects.

5 DISCUSSION

Our results reflect the opinion of companies operating in different countries, dealing with different domains, and using different process models and tools. The results further the current RE community effort to move towards a problem-driven research agenda, where solutions are discussed and developed together with industry [6]. In this section, we report the main implications of our work together with its limitations.

5.1 Implications

The implications are based on the results presented in Section 4 as well as the state-of-practice reported in Section 2. We suggest three focus areas for designing a platform to support large-scale RE.

Strategies to fight back information overload. Out of the identified problems, as can be expected for companies dealing with large-scale requirements, the main one is information overload. Accordingly, the causes for the other

pain-points reported can be traced back to the amount and diversity of data the stakeholders need to deal with, as their tasks (e.g., requirements dependency or stakeholder identification) become harder due to the scale of the data. However, information overload does not appear among the problems currently perceived in the current RE state-of-practice [7]. This observation holds even when only companies operating at a similar scale to the one considered in this study are taken into account. On the other hand, fighting information overload is one of the research areas identified by Regnell *et al.* in the context of large- and very large-scale RE [25].

Directly related to this problem is what Wnuk *et al.* [34] call “scoping”—i.e., a set of mechanisms to cope with requirements scalability problems. Although not explicitly mentioned during the interviews, “scoping” is what the companies would like to achieve through better data analytics solutions. Our results showed two specific instances in which data analytics should be applied to cope with information overload, visualization, and effort estimation. Visualization is perceived as a way to see the “big picture”—a need already expressed in [34]; whereas effort estimation can help to realize a more predictable process when lots of requirements need to be handled. Although effort estimation is well-studied in software development (in open source [33] or otherwise [32]), few studies address it from an RE perspective despite other industrial studies pointing to its importance (e.g., [14]). In practice, trying to control information overload through effort estimation does not seem to be actively pursued as the primary mechanisms to fulfill such need (i.e., collecting the necessary metrics to calculate effort) are not in place. The companies did not explain why that may be the case.

The fear of complicating the process and the selection of the sources for data analytic approaches are strictly related. On the one hand, it is not recommended to add new data which is likely to not bring improvement while impacting the process. On the other hand, discarding a data source can introduce a bias in the analytic. Such decision should be evaluated in the context of the company. For example, in the case of Company A, such scenario is sensible due to the level of public accountability of the company towards its community. We recommend further investigations on information overload and scoping in large-scale RE to focus on the feasibility of effort estimation techniques. Moreover, trade-offs—based on the company context—need to be taken into account.

Stakeholder role in requirements dependency and reuse. Both requirements dependency and reuse were identified as pain-points, supporting the results from previous exploratory studies [22, 23]. However, a new factor emerged in our study, the importance of different stakeholders in developing these approaches. On one hand, not only artifact-based features (e.g., the project to which a requirement belongs to) but also the requirement engineer personal preferences need to be taken into account when defining the dependencies. On the other hand, attention should be given to the fact that other stakeholders should be able to interpret the dependencies.

Previous research has pointed out the “people” factor as a barrier for the adoption of requirement reuse. For example, Palomares *et al.* [22] reports the resistance to change of requirements engineers as the most prominent of such barriers. Instead, the challenge we observed is the loss of knowledge, necessary for reuse, due to employees turnover. In turn, this observation points to the importance of knowledge maintenance for requirements reuse in line with the results presented in [12, 22] and, in general, supports the recommendation of Dyba [5] for software process improvements. We report volatility of requirements as an additional challenge to reuse. This further supports the result of Sethia *et al.* [29] which identifies such factor as one of the most prominent reasons for project failures.

The results of this study remark the importance of including stakeholders in the definition of requirements dependencies and reuse. For the former, particular attention should be given to configurability and interpretability; for the latter focus should be given to manage both endogenous (i.e., within the company) and exogenous (i.e., from external factors) changes.

Integration of stakeholder in tool-based decisions. From our results, it emerged that RE tools are needed to create a holistic view of the decisions about requirements. However, to reach such objective, integration should be as easy as possible. Nevertheless, Camarillo deGea *et al.* [4] shows that only a few tools support open exchanging formats, such as ReqIF. This is needed especially for vertical integration, since companies work on different requirements repositories according to their business units and with different tools. Focus on the “people” factor is also necessary for better tooling. Maalej *et al.* [16] shows that the requirement author intention is one of the top information needs sought by a stakeholder when understanding a requirement, and at the same time, it is one of the least supported features in current tools. Accordingly, our study shows the need for stakeholders to incorporate their feedback as part of the results presented by a tool. Such mechanism can improve the transparency of the decision-making process, thus providing first support for resolving conflicts between stakeholders—i.e., the other information need currently ill-supported by RE tools [16].

5.2 Limitations

In this section, we briefly summarize the limitations of our case study based on the guidelines presented in [27].

However, as interviews were the only data collection method, we could have missed relevant information.

Construct validity. This threat can arise once there is not a shared understanding of the terms used in the interviews between interviewers and interviewees. In our study, the interviews were based on concepts from the domain of the stakeholders who participated in the study. The company representatives were involved in the creation of the interview script. At the beginning of each interview session, we dedicated time to go through the status quo of the company with the participants. In addition, some of the interviews were carried out with managers and some of their employees. In our study, we believe this is no big issue since: 1) since the companies where the interviews where this happened do not follow a hierarchical employee structure (i.e., there is no fear to managers), and 2) The employees were really participative in the interviews. However, we cannot control some information was missed out because of that. Triangulation with other data collection methods (e.g., observation) could have improved construct validity.

Internal validity. This study does not involve the assessment of causal relationships using statistical methods, as the case study is purely descriptive. The recommendation presented in Section 5.1 are based on our interpretation of the codes, their co-occurrence, and their importance as perceived by the participants. The themes emerged after at least two researchers reviewed the codes.

External validity. As it is usual for case studies, we do not claim strong generalization of our findings. We showed that the results apply to different extent within the case companies. We make this explicit when reporting the findings by indicating the context in which they are more sensible. Only some of our results (e.g., the pain-point of information overload) generalize over the three cases. For these results, we do not claim statistical generalization to a pre-defined population of companies due to the low number of cases and the non-probabilistic sampling approach driven by the project settings. However, they are important in the context of the project to, for example, prioritize what needs should be fulfilled when developing a solution. We claim analytical generalization of our findings by expanding the theory current on RE-related pains and needs presented in Méndez *et al.* [7] when considering companies dealing with large-scale requirements in a bespoke or market-driven domain.

Reliability. We strengthened the reliability of the interview scripts by running an internal pilot; moreover, we used established coding techniques and tools to code the collected data. On the other hand, we acknowledge that the researchers, as well as some of the interviewees, had in-depth knowledge on the context in which the case studies

were carried out—e.g., the OPENREQ project. Therefore, other researchers wanting to carry-out the same study need to familiarize with the background and objectives of the project.

6 CONCLUSION

In this paper, we report the results of a multiple case study with three companies dealing with large-scale RE. The interviews with ten requirement engineers indicate information overload, requirements dependency identification, and current tool limitations as the main pain-points for individual stakeholders. Accordingly, a platform to support large-scale RE should address the need of gathering data through existing tools integration to offer analytic solutions. A sensible RE task for such data pipeline is effort estimation. Moreover, we showed that stakeholders preferences must be included when devising approaches for requirements dependency and reuse patterns identification. Similarly, the platform should take into account stakeholders feedback and trace decision-making from upstream in the RE process. Finally, we identified several challenges for the adoption of such a platform.

We will further validate our findings through focus groups and confirmatory surveys. Finally, we will evaluate a prototype version of the platform with the companies involved in this study.

ACKNOWLEDGMENTS

The authors of this paper have received funding from the European Union Horizon 2020 research and innovation Programme under grant agreement No 732463.

REFERENCES

- [1] Daniel M. Berry. 2017. Evaluation of Tools for Hairy Requirements and Software Engineering Tasks. *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW) (2017)*, 284–291.
- [2] Kathy Charmaz. 2006. *Constructing Grounded Theory: A Practical Guide through Qualitative Analysis (Introducing Qualitative Methods series)*. SAGE Publications.
- [3] Daniela Damian and James Chisan. 2006. An empirical study of the complex relationships between requirements engineering processes and other processes that lead to payoffs in productivity, quality, and risk management. *IEEE Transactions on Software Engineering* 32, 7 (2006), 433–453.
- [4] Juan M Carrillo De Gea, Joaquín Nicolás, José L. Fernández Alemán, Ambrosio Toval, Christof Ebert, and Aurora Vizcaíno. 2012. Requirements engineering tools: Capabilities, survey and assessment. *Information and Software Technology* 54, 10 (2012), 1142–1157.
- [5] Tore Dybå. 2005. An empirical investigation of the key factors for success in software process improvement. *IEEE Transactions on Software Engineering* 31, 5 (2005), 410–424.
- [6] Daniel Méndez Fernández and Stefan Wagner. 2015. Naming the pain in requirements engineering: A design for a global family of surveys and first results from Germany. *Information and Software Technology* 57 (2015), 616–643.
- [7] Daniel Méndez Fernández, Stefan Wagner, Marcos Kalinowski, Michael Felderer, Priscilla Mafra, Antonio Vetrò, Tayana Conte, Magne Christiansson, Desmond Greer, Casper Lassenius, et al. 2017. Naming the pain in requirements engineering. *Empirical software engineering* 22, 5 (2017), 2298–2338.
- [8] Daniel Méndez Fernández, Stefan Wagner, Klaus Lochmann, Andrea Baumann, and Holger de Carne. 2012. Field study on requirements engineering: Investigation of artefacts, project parameters, and execution strategies. *Information and Software Technology* 54, 2 (2012), 162–178.
- [9] Donald Firesmith. 2004. Prioritizing requirements. *Journal of Object Technology* 3, 8 (2004), 35–48.
- [10] Xavier Franch, Daniel Méndez Fernández, Marc Oriol, Andreas Vogelsang, Rogardt Heldal, Eric Knauss, Guilherme Travassos, Jeffrey Carver, Oscar Dieste, and Thomas Zimmermann. 2017. How do Practitioners Perceive the Relevance of Requirements Engineering Research? An Ongoing Study. In *2017 IEEE 25th International Requirements Engineering Conference (RE)*. 382–387. <https://doi.org/10.1109/RE.2017.17>
- [11] Heli Hiisilä, Marjo Kauppinen, and Sari Kujala. 2015. Challenges of the Customer Organization’s Requirements Engineering Process in the Outsourced Environment—A Case Study. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer, 214–229.
- [12] Axel Hoffmann, Andrej Janzen, Holger Hoffmann, and J Leimeister. 2014. Success Factors for Requirement Patterns Approaches Exploring Requirements Analysts’ Opinions and Whishes. (2014).
- [13] Hubert Hofmann and Franz Lehner. 2001. Requirements engineering as a success factor in software projects. *IEEE software* 18, 4 (2001), 58.
- [14] Sofija Hotomski, Eya Ben Charrada, and Martin Glinz. 2016. An exploratory study on handling requirements and acceptance test documentation in industry. In *Requirements Engineering Conference (RE), 2016 IEEE 24th International*. IEEE, 116–125.

- [15] Timo Johann, Christoph Stanik, Alizadeh Alireza, and Walid Maalej. 2017. SAFE: A Simple Approach for Feature Extraction from App Descriptions and App Reviews. *2017 IEEE 25th International Requirements Engineering Conference (RE)* (2017), 21–30.
- [16] Walid Maalej, Zijad Kurtanovic, and Alexander Felfernig. 2014. What stakeholders need to know about requirements. In *Empirical Requirements Engineering (EmpiRE), 2014 IEEE Fourth International Workshop on*. IEEE, 64–71.
- [17] Walid Maalej, Maleknaz Nayebi, Timo Johann, and Guenther Ruhe. 2016. Toward data-driven requirements engineering. *IEEE Software* 33, 1 (2016), 48–54.
- [18] Michele Marchesi, Giancarlo Succi, Don Wells, Laurie Williams, and James Donovan Wells. 2003. *Extreme programming perspectives*. Vol. 176. Addison-Wesley.
- [19] Tommi Mikkonen, Casper Lassenius, Tomi Männistö, Markku Oivo, and Janne Järvinen. 2018. Continuous and collaborative technology transfer: Software engineering research with real-time industry impact. *Information and Software Technology* 95 (2018), 34–45.
- [20] Matthew B. Miles and A. Michael Huberman. 1994. *Qualitative data analysis: An expanded sourcebook*. sage.
- [21] Cristina Palomares, Xavier Franch, and Davide Fucci. 2018. Personal Recommendations in Requirements Engineering: The OpenReq Approach. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer, 297–304.
- [22] Cristina Palomares, Carme Quer, and Xavier Franch. 2017. Requirements reuse and requirement patterns: a state of the practice survey. *Empirical Software Engineering* 22, 6 (2017), 2719–2762.
- [23] Mikko Raatikainen, Tomi Männistö, Teemu Tommila, and Janne Valkonen. 2011. Challenges of requirements engineering—A case study in nuclear energy domain. In *Requirements Engineering Conference (RE), 2011 19th IEEE International*. IEEE, 253–258.
- [24] Björn Regnell and Sjaak Brinkkemper. 2005. Market-driven requirements engineering for software products. In *Engineering and managing software requirements*. Springer, 287–308.
- [25] Björn Regnell, Richard Berntsson Svensson, and Krzysztof Wnuk. 2008. Can we beat the complexity of very large-scale requirements engineering?. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer, 123–128.
- [26] Patrick Rempel, Patrick Mçder, and Tobias Kuschke. 2013. An empirical study on project-specific traceability strategies. In *Requirements Engineering Conference (RE), 2013 21st IEEE International*. IEEE, 195–204.
- [27] Per Runeson and Martin Höst. 2009. Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering* 14, 2 (2009), 131.
- [28] Carolyn B. Seaman. 1999. Qualitative methods in empirical studies of software engineering. *IEEE Transactions on software engineering* 25, 4 (1999), 557–572.
- [29] Neetu Kumari Sethia and Anitha S Pillai. 2014. The effects of requirements elicitation issues on software project performance: An empirical analysis. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer, 285–300.
- [30] Ernst Sikora, Bastian Tenbergen, and Klaus Pohl. 2011. Requirements engineering for embedded systems: An investigation of industry needs. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer, 151–165.
- [31] Muhammad Usman, Emilia Mendes, and Jürgen Börstler. 2015. Effort estimation in agile software development: a survey on the state of the practice. In *Proceedings of the 19th International Conference on Evaluation and Assessment in Software Engineering*. ACM, 12.
- [32] Muhammad Usman, Emilia Mendes, Francila Weidt, and Ricardo Britto. 2014. Effort estimation in agile software development: a systematic literature review. In *Proceedings of the 10th International Conference on Predictive Models in Software Engineering*. ACM, 82–91.
- [33] Cathrin Weiss, Rahul Premraj, Thomas Zimmermann, and Andreas Zeller. 2007. How long will it take to fix this bug?. In *Mining Software Repositories, 2007. ICSE Workshops MSR'07. Fourth International Workshop on*. IEEE, 1–1.
- [34] Krzysztof Wnuk, Björn Regnell, and Brian Berenbach. 2011. Scaling up requirements engineering—exploring the challenges of increasing size and complexity in market-driven software development. In *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer, 54–59.
- [35] Robert K Yin. 2017. *Case study research and applications: Design and methods*. Sage publications.