BibTeX

Challenges of Requirements Communication and Digital Assets Verification in Infrastructure Projects

Waleed Abdeen*[iD], Krzysztof Wnuk[iD], Michael Unterkalmsteiner[iD], Alexandros Chirtoglou

*Corresponding author: waleed.abdeen@bth.se

Article info

Abstract

Background: Poor communication of requirements between clients and suppliers contributes to project overruns,in both software and infrastructure projects. Existing literature offers limited insights into the communication challenges at this interface.
Aim: Our research aim to explore the processes and associated challenges with requirements activities that include client-supplier interaction and communication.
Method: we study requirements validation, communication, and digital asset verification processes through two case studies in the road and railway sectors, involving interviews with ten experts across three companies.
Results: We identify 13 challenges, along with their causes and consequences, and suggest solution areas from existing literature.
Conclusion: Interestingly, the challenges in infrastructure projects mirror those found in software engineering, highlighting a need for further research to validate potential solutions.

1. Introduction

Developing software products and releasing them to the market is a complex process that requires efficient communication [1]. Inefficient communication manifests itself as communication gaps that lead to quality issues, wasted efforts, delays, and ultimately to failure to meet the customers' expectations [2]. Moreover, the inefficient use of requirements artifacts, i.e., information is not tailored to the needs of the involved stakeholders, may impede requirements communication in software projects [3]. What is common for large software and infrastructure projects is their complexity that often implies the use of suppliers [4] to deliver significant parts of the solution, e.g. automotive industry often uses suppliers to develop software for cars.

The friction-free economic principles of the software industry help involve suppliers from geographically distant locations, primarily for cost reduction and the availability of competence. One of the main challenges in involving suppliers is establishing good communication principles, especially agreed quality levels, and also communicating potential difficulties and issues early [5]. In particular, the verification of supplier deliverables against

1

client requirements poses challenges [6]. A large contributor to this challenge is the inherent uncertainty in requirements, which increases the negative influence of coordination efforts on trust for new product development projects [7]. In project management theory, it has been suggested that both goal and resource interdependence between clients and suppliers have a positive influence on coordination and cooperation in new software development programs [8].

Infrastructure projects often last decades in planning and execution [9], and involve a large number of specialized project teams (e.g., owners, contractors, consultants, designers) with specific core competencies. These projects are also featured with uncertainty, fragmentation [10], and high complexity and inter-organizational task interdependence, which makes communication ever more important [11]. Because contracts of construction projects are inherently incomplete [12], understanding which challenges arise in client-supplier communication in design-build projects becomes critical.

In practice, the ability to communicate clients' needs and requirements to suppliers is a key success criterion in infrastructure projects [13–15]. Effective communication facilitates stakeholder engagement during the change management process [9] and clarifies realistic stakeholder expectations, while insufficient communication drives uncontrolled change and can contribute to project delays [16].

Several studies have explored the challenges associated with general communication in large construction projects [17], requirements allocation [18], verification and validation [19], system integration [20], the relationship between the communication-conflict interaction and project success [10], trust between firms and suppliers [21], encoding and decoding communication competencies in project management [22], or inter-cultural communication [23].

However, little research effort has been dedicated to exploring the processes and associated challenges with requirements activities that include client-supplier interaction and communication in infrastructure projects. Requirements communication is the process of communicating — both verbally and through documents — requirements and project-related documents between the client and the supplier. Requirements validation is the process of ensuring that the requirements reflect the client's needs before designing or investing in building the system. Digital assets verification ensures that the delivered assets (e.g., 3D drawings) correspond to the requirements specifications. Figure 1 illustrates the verification and validation processes. It is important to note that in the verification processes (requirements verification and system verification), the client is not directly involved. As for the validation processes, the client input is important to ensure delivery of what the client wants. In infrastructure projects, tender documents, written by the client, are the basis from which suppliers derive requirements specifications that need to be validated.

This study is part of a research project that aims to investigate methods that can improve functional requirements traceability at different stages in construction projects. Functional requirements should be linked to a digital twin [24], a collection of digital assets that represent the facility and are linked to the physical facility. Digital assets are defined as any digital file (textual, images, audio, or video) stored on any electronic device (e.g., computer, mobile phone, or cloud) with the right to own the file [25]. In infrastructure projects, Computer-Aided Design models (CAD), Building Information Models (BIM), and System Information Models (SIM) [26] are typical digital assets. Traceability between functional requirements and digital assets would allow for efficient and effective digital assets verification before construction has started, enable monitoring during construction,
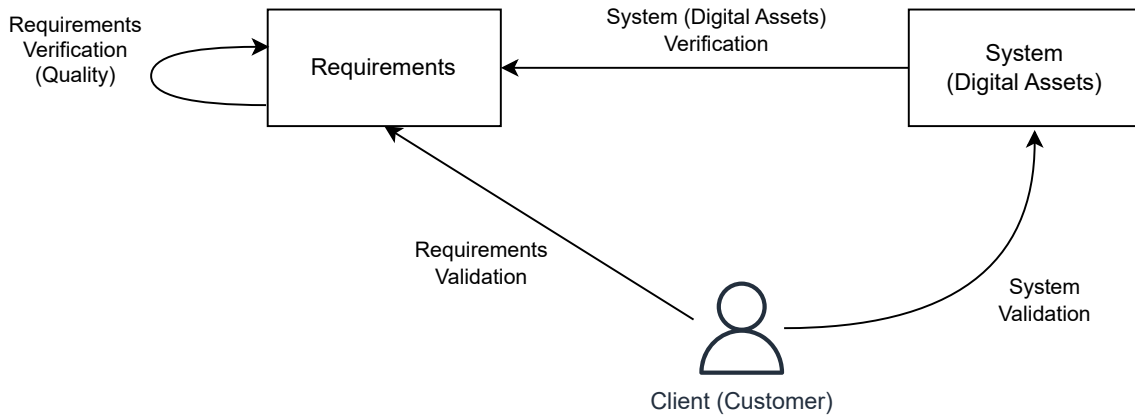
Figure 1. Verification and Validation Processes

as well as make it possible to follow-up on the original requirements during the decades of operation and maintenance.

This study explores the nature of client-supplier requirements communication in large infrastructure design-build projects, where the supplier is responsible for the design, construction, and requirements validation and digital assets verification processes. Such a description can help to understand the challenges that practitioners encounter and either point to solutions that exist in the literature or describe a research gap.

We frame our research using the design science process [27, 28], which aims to extend knowledge by identifying and addressing specific problems in practice. We conducted two case studies where we explored the client-supplier interface. In the two projects, Anonymous acted as client and two other companies acted as contractor and sub-contractor. The semi-structured interviews were designed to understand *how* requirements communication, requirements validation, and the verification of deliverables are conducted. We transcribed the audio recordings and coded the data to answer the research questions.

We present an overview of the client-supplier interface and develop process diagrams for each investigated case. Furthermore, we identify and describe the challenges faced by the managers and engineers working in these processes. Finally, we map these challenges to potential solutions from the system and software engineering domain. *Identify conflicts early*, *requirements abstraction*, *requirements are impossible to build* and *granularity of traces*, are challenges that are also faced in software engineering domain and are important to address.

The remainder of this paper is structured as follows. We present in Section 2 literature that is related to our work. In Section 3, we present our research methodology and study design. We present our research results in Section 4 and discuss them in Section 5. In Section 6, we draw a conclusion from this study. And finally we list future work directions in Section 7.

2. Related Work

The focus of our investigation is the processes related to requirements engineering on the interface between clients and suppliers in the context of construction projects. In this section, we discuss, therefore, related work from systems engineering and communication in large construction project research areas.

2.1. Communication in Software Engineering

Communication in software engineering is essential in any project as it affects its outcome. Many researchers have investigated the associated challenges with communication in industry [3, 29, 30]. Bjarnason et al. [29] investigated geographical, cognitive, and psychological distances in communication between teams and identified ten factors that affect communication. Liskin [3] investigated how artifacts support or impede requirements communication. They identified five challenges related to link requirements. Liebel et al. [30] studied communication problems in automotive requirements engineering, concluding that it is important to establish communication channels outside the fixed organization structure. Other researchers [31–33] proposed solutions to tackle these challenges. Iqbal et al. [31] developed a framework to address communication issues during the requirements engineering process for software development outsourcing. Fricker et al. [32] suggested handshaking with implementation proposals that could be used to improve the communication between the client and the supplier. Pernstal et al. [33] developed the BRASS framework to support coordination and communication of inter-departmental requirements in the large-scale development of software-intensive systems. However, the framework does not focus on requirements validation but rather on supporting the selection of the most suitable implementation plan. Although these studies investigate (requirements) communication challenges and propose solutions to them, they mainly focus on inter-organization communication, in contrast to our study, where we focus on requirements communication in the client-supplier interface.

2.2. System Engineering

RE and verification are of high interest to the system engineering community, which can be seen by the several conducted studies covering different system engineering processes and their challenges [18, 20, 34–36].

Madni et al. [20] explored the challenges in system integration in the defense and aerospace domain. They propose an ontology-based system integration approach to facilitate communication between the different stakeholders. Makkinga et al. [19] conducted semi-structured interviews in three mid-sized projects in the Netherlands. Their focus was on the verification and validation problems in construction projects and possible solutions to these problems. They advise more research work in the validation process. Vermillion et al. [18] compare the requirements allocation and objective allocation in projects with outsourced design. They propose an approach for requirements allocation for the design process. Moreover, they see an asymmetry in knowledge between the clients who outsource the design and agents designing the system, which needs to be addressed to reduce negotiation iterations.

System engineering processes and practices in the construction domain have also been of interest to researchers [35, 36]. Lynghaug et al. [35] explored the state of system engineering practices in the Norwegian construction domain. They conclude by providing recommendations on implementing and improving system engineering practices in the construction domain. One of the most critical system engineering processes is requirements analysis. Raatikainen et al. [37] highlighted the challenge of efficient communication and management of requirements in the nuclear energy domain. De Graaf et al. [36] investigated the state of practice of six cases of sub-contracted design work in civil engineering projects. They identified three factors that affect the sub-contracted work,

mainly: Building Information Model (BIM) interoperability, time pressure, and employee availability.

2.3. Communication in Large Construction Projects

In large construction projects, communication is essential to the success of the project, and conflicts could lead to communication breakdowns and project failure as illustrated in multiple studies [9, 10, 17, 38]. Wu et al. [10] investigated the relationship between the communication-conflict interaction and project success. Malik and Taqi [17] explored the relationship between communication and the success of construction projects. They found that process conflict and relationship conflict have a negative impact on communication and project success. Saxena and McDonagh [38] have investigated communication breakdowns and concluded that change management requires a multilevel communication approach. Furthermore, communication could be an enabler to other activities in the project. Butt et al. [9] have looked into how effective communication facilitates stakeholder engagement during the change management process and changing project culture. Communication is a powerful tool in ensuring participation in change management processes. Lack of communication leads to teams focusing on task performance and efficiency rather than empowerment and involvement.

Communication is even more essential in global projects, where cultural changes added more complexity, as it continues to shows [22, 23, 39–41]. Loosemore and Muslmani [23] investigated intercultural communication challenges in Persian Gulf projects, highlighting the need for a better understanding of cultural diversity. Similar recommendations were reported by Ochieng and Price, who studied the cultural variation of project managers in Kenya and the UK in communicating effectively on multicultural projects [39]. Henderson et al. [41] looked into the impact of communication norms on global project teams and individuals' project satisfaction and performance. In another study [22], the researchers also looked into encoding and decoding communication competencies in project management. Daim et al. [40] have also looked into communication breakdowns among global virtual teams, and these breakdowns tend to threaten project delivery. The authors found five factors impacting communication breakdowns: trust, interpersonal relations, cultural differences, leadership and technology.

These studies show the importance and consequences of communication in construction projects. However, they are mainly focused on people's communications rather than artifacts of communication, which we focus on in our study.

2.4. Research Gap

Requirements validation, communication and digital assets verification are important in avoiding project failures [15, 42]. While researchers from the software engineering domain have investigated the area [43, 44], it remains greatly unexplored or neglected in the system engineering domain. Hence, due to the lack of studies that investigate requirements validation, requirements communication and digital assets verification in the system engineering domain, we are conducting this study to fill this gap.

3. Methodology

Due to the aim of the research project as a whole, which is to identify and address problems related to the requirements communication process and the traceability of functional requirements to the digital twin, we frame our research as design science problem [27, 28]. Hevner [45] has depicted design science as a process intertwining relevance, rigor, and design cycles. In the study reported in this paper, we investigate the challenges of client-supplier communication in system engineering projects; in other words, we identify the *relevance* of the problem in the infrastructure domain. Thereafter, we plan to design a solution for one of the problems we identify in this study, apply it to the problem, and further improve it (*design*). Finally, we plan to verify that the solution fits the problem through implementation to other problem instances from the software engineering domain (*rigor*). In this study, we focus only on determining the *relevance* of the problem.

Table 1. Research questions

Id	Research Question	Motivation
RQ1	What are the practices and challenges in system requirements validation?	One of the main documents that are communicated between the client and the supplier in design-build contracts are system requirements. Hence it is important to have those requirements validated. We explore the system requirements validation process and the challenges associated with it.
RQ2	What are the practices and challenges in system requirements communication between the client and the supplier?	Explore the requirements communication process between the client and the supplier, and what different formats the requirements take before they are translated into a design. Moreover, we want to explore the difficulties in the process and opportunities for improvements.
RQ3	What are the practices and challenges in digital assets verification?	Explore any quality checks done by the client or the supplier on the design documents and any challenges associated with the process.

We designed the study according to the case study guidelines by Runeson et al. [46], who advocate for defining a detailed case study protocol that reflects the changes made during the iterative process of data collection and analysis. The research questions with the motivation and alignment to our objective are listed in Table 1.

3.1. Case description

The two cases were selected from *Anonymous'* projects based on availability and access to information. Both projects distinguish between project-specific requirements and regulatory requirements. Project-specific requirements describe needs that pertain to the particular facility, originating from diverse stakeholders such as the government, communes, and land owners. Regulatory requirements define needs that are relevant to all applicable facilities. While the client specifies the project-specific requirements, it is the responsibility of the supplier to identify and comply with the relevant regulatory requirements for the designed facility.

Case One is a road project that includes the design and building of roads and bridges for cars, pedestrians, and cyclists. At the time of the study (April-June 2020), the project was in construction (3 years duration). Case One, with a budget of 35M USD, is part of

Table 2. Companies overview

Company	Industry	Role	Case	Company Size
A	Construction	Contractor	Case One	45,000
B	Design	Sub-Contractor	Case One	6,000
C	Transportation and Infrastructure	Client	Case One, Case Two	9400

The unit of measure for company size is the average number of employees

Table 3. Interviewees roles

Id	Role	Company	Experience
A1	Design Manager	A	20+ years
A2	Tender Manager	A	10 years
A3	BIM Manager	A	7 years
A4	Design Manager	A	10 years
B1	Discipline Leader	B	15 years
B2	Head of Design	B	25 years
C1	BIM Specialist	C	9 years
C2	Requirements Specialist	C	11 years
C3	Requirements Engineer	C	8 years
C4	Head of Tech and Environment	C	24 years

The experience presented in this table is the overall industry experience.

a larger road project with an estimated duration of 30 years (15 planning, 15 construction) and a budget of 4B USD. Case Two has specified approximately 700 project-specific requirements and is associated with 12,000 regulatory requirements.

Case Two is a railway project with the design and build of a high-speed train connection, including rail, bridges, and tunnels. The project was in its early stages (requirements specification and initial design) at the time of the study. The estimated project duration for Case Two is 34 years (9 pre-study, 10 planning, 15 construction) and a budget of 8.8B USD. Case Two has specified approximately 1,000 project-specific requirements and is associated with approximately 15,000 regulatory requirements[1].

3.2. Data collection

We conducted semi-structured interviews with the participants from all three companies listed in Table 2. Semi-structured interviews are fit for a study where a clear hypothesis does not exist and the research questions are of explorative nature [46]. In our case, we explored the system engineering processes that involve client-supplier communication, but we don't know to what extent these processes were implemented.

We formulated the interview questions in three main themes, according to our stated research questions: *requirements validation*, *requirements communication*, and *digital assets verification*. Then we adapted the interview questions to each interviewee role. For example, when we interviewed a requirements engineer, we focused our questions on *requirements validation* and *requirements verification*. During each interview, we asked (on average) 14 pre-defined questions about the topics, excluding follow-up questions. We present here a few examples of the questions that we asked during the interviews: on the topic of *requirements*

---

[1]In both projects, the number of project-specific requirements varied over time. Furthermore, the number of associated regulatory requirements is an estimate by a requirements engineering lead at *Anonymous*. Determining the exact numbers would require that every contractor keeps track of the regulations they need to comply to, which is, as we shall see, not the case.
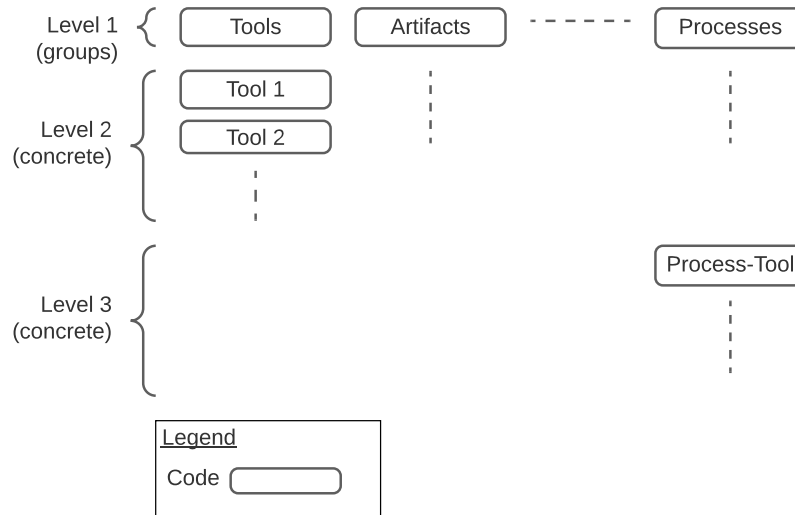
Figure 2. Coding levels

communication we asked questions such as "Q: How do you receive the requirements specified by the client?" and "Q: Do you experience any challenges in requirements' communication?", on the topic of *requirements validation* we asked "Q: Explain the requirements validation process?", and in the area of *digital assets verification* we asked "Q: What is the next step after the digital assets are produced?", and "Q: How do you ensure that the produced digital assets satisfy the requirements".

We interviewed ten people from three companies. Table 3 shows their roles, companies, and industry experience. The interviewees had between 7 and 24 years of experience and worked in roles related to requirements engineering, design, and project management. Convenience sampling was used when choosing the participants for the interviews. We asked our company contacts for people who work with requirements and system design, then they suggested people based on their availability. In Case 2, we only interviewed people from the client side as we didn't have access to suppliers. In this paper, we use the acronym XY to reference interviewees, where X is a letter referencing the company [A,B,C] and Y is a number referencing an interviewee in that company.

Two to three researchers attended the interviews (observer triangulation). The first author led the interview by asking questions, and the other two authors observed and asked follow-up questions if necessary. We conducted each interview as follows:

1. A week before the interview, we sent the informed consent letter to the participants and asked them to return a signed copy within two weeks after the interview.
2. Before the interview started, we briefly described the purpose of our study, and the expected outcome and explained how we are going to conduct the interview.
3. We started recording the interview and asked the questions.

After we finished conducting the interviews, the first author transcribed them. The third author listened to the interviews' recordings to verify the transcript. We sent each transcript out to the interviewees for comments. We received corrections from one participant, clarifying statements that were incomplete due to recording issues caused by the recording equipment.

3.3. Data analysis

We coded the interview transcripts with an initial set of codes that the first and third author created in alignment with the research questions. The first author quoted and labeled parts of the transcript using these codes. Feedback sessions were conducted between the first and third author to refine the codes. Several coding iterations were done until we had enough data to answer the research questions. A final verification of the quoted text was done by the third author. The codes were created on different levels of abstraction, as seen in Figure 2. Initially, we started with 14 abstract codes: *artifacts, roles, tolls, processes, important statements and challenges, digital assets implementation, digital assets verification, requirements elicitation, requirements specifications, requirements communication, requirements change, and requirements validation.* We derived compound codes by combining two or more abstract codes (e.g., requirements validation-process). Then, we added additional codes as needed. This helped us to use relevant codes when answering the research questions, as exemplified next.

Coding example When coding the answers to the interview questions related to the requirements validation process, we started with the codes: 1) *requirements validation process* to quote the activities associated with the requirements validation process and 2) *requirements validation challenge*, to quote the challenges of that process. These codes were part of the group *requirements validation*. Then we added more codes based on the content, e.g., *requirements documentation* and *cost estimation* to quote the artifacts used in the validation process. After that, we created codes on a higher abstraction level e.g., *tools*, *artifacts*, and *processes*.

Using this way of coding resulted in codes on different abstraction levels. This helped retrieve all information in a whole area, e.g., the requirements validation process, and in detail in that process, e.g. actors in the requirements validation process.

4. Results

We present the results in three sections following the themes of the interview questions: requirements validation, requirements communication, and digital assets verification. In each section, we describe the process and the challenges faced in that process. A challenge is presented as the description, causes, and consequences of that challenge. Some challenges have no causes or/and consequences since we could not identify them from the interviews. We illustrate the results in figures that show the process flow, artifacts, and actors. Each figure is divided into columns representing the parties involved in the project: client, contractor, or subcontractor. In some figures, there is a question mark which is a placeholder for information that was not clear from the interviews.

4.1. Requirements validation

Figure 3 depicts the requirements validation process in Case One. At the contractor, the tender manager and their team conduct a review process. The inputs to the process are the technical description and the general regulations (local to the country) that apply to the project. The output of the process is a cost estimate spreadsheet, which estimates the project cost and is used when preparing the contract for the project. Another requirements review process is conducted at the subcontractor by the discipline leaders and designers.
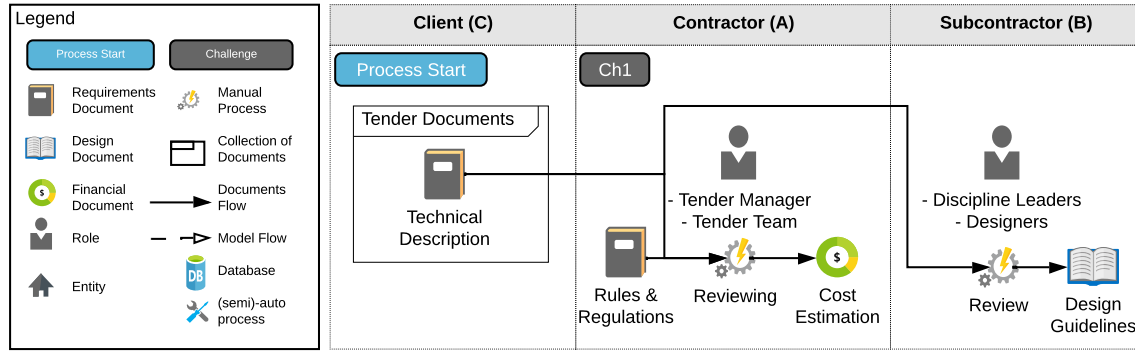
Figure 3. Requirements validation – Case one, the abbreviation (ch) refers to challenge, both the abbreviations and legend apply to all figures
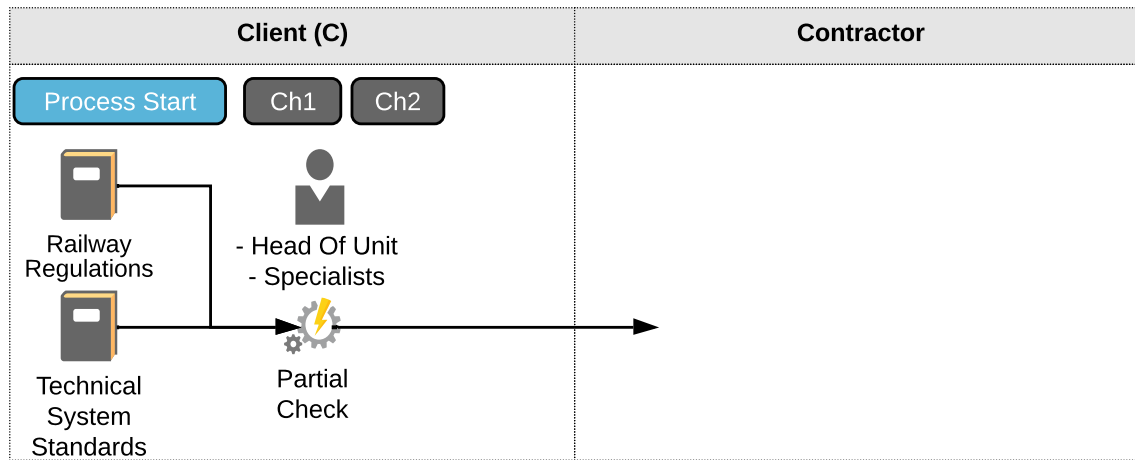

Figure 4. Requirements validation – Case two

They take as input the technical description from the contractor and the general regulations. The output of this process is the design guidelines document that helps designing the models for the project.

Figure 4 shows the requirements validation process in Case Two. The client conducts a partial check on the requirements. The inputs to this check are the railway regulations and technical system standards and the outputs are the validated requirements. We do not have information about the requirements validation process on the contractor side, because the people we interviewed in Case Two are from the client-side.

We have identified the following two challenges that are faced by the roles involved in the requirements validation process.

4.1.1. Challenge 1: Prioritizing the requirements validation process

Prioritizing the requirements validation process is a challenge faced by the contractor in Case One and the client in Case Two. The time to validate the requirements by the contractor is seen to be short in Case One, as reported by the tender manager (A2). When the tender project is announced, the contractor has to respond with the cost estimate in a matter of weeks. This makes the validation of the requirements challenging for the contractor who has to do it in a short time. "The client spends four years coming up with the design (presented by the requirements), and we have to price it in four or eight

weeks, coupled with a tender model that is purely price-driven. It is a very unhealthy situation" (A2). The requirements validation process requires experience and knowledge in the discipline, which makes the prioritization of this process and finding the right people to perform it difficult for the client. "When you talk about validating requirements, it is not our top priority" (C2).

Ambiguous and conflicting requirements is a cause of this challenge in Case One. The requirements validation process could take more than a couple of months when the supplier finds requirements that need clarification or revision by the client. In Case Two, *lack of resources* is seen as a cause of this challenge. As C2 said, "there are not enough railway specialists in the company." The people involved in Case Two are involved in other projects and have other responsibilities at the client, making it difficult to find the right people to validate the requirements.

The requirements validation process is overlooked is seen by the client as a consequence of not prioritizing the said process. Since it is not always feasible to allocate resources for the validation process, it gets less priority over other activities, leading to the process being skipped in many cases. Furthermore, the contractor sees *added responsibility* as a consequence. Since the contractor submits the bid to the project based on the requirements, they take the responsibility if they have not validated the requirements properly: "Do not put all the risk on us if we can't find the problem in such a short time" (A2).

4.1.2. Challenge 2: Identify conflicts early

It could be difficult to identify conflicts in the requirements early as seen in Case Two by the requirements engineer (C3) and head of environment (C4). Although a partial check on the requirements is conducted at the client side, some conflicts may go undetected until later at the design stage. For example, when verifying tunnel-related requirements, one (requirements engineers or designers) may not be able to identify water pipes intersection with an obstacle due to the land geometry. "It is hard to see that before when you haven't drawn the lines for the pipes or the tunnel" (C3).

The main cause of this challenge is not clear. When we asked C3 whether the lack of information or having too much information is the cause of difficulties identifying those conflicts, they answered "it depends" (C3).

4.2. Requirements communication

Figure 5 shows the requirements communication process in Case One. The project's technical requirements, called technical description document, are a part of the tender documents. The project leaders and tender manager on the contractor side get the technical description from the client, and then they communicate these documents to the discipline leaders and designers on the subcontractor side. Additional requirements from the rules and regulations apply to the project in Case One. The contractor and subcontractor are responsible for finding those additional requirements that apply to the project.

Figure 6 shows the requirements communication process in Case Two. In this case, there are three main types of requirements sources: 1) documents, e.g., technical system standards, or railway regulations; 2) internal or external departments, e.g., local government or maintenance department; and 3) people who reside or work in the project area and who could be affected by the project outcome, e.g., property owners. The client compiles the requirements from all the mentioned sources into project-specific requirements for the
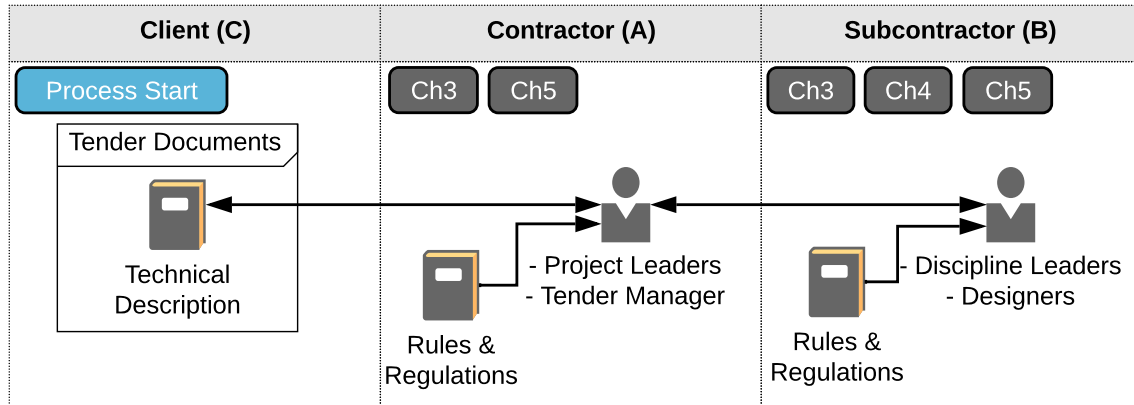
Figure 5. Requirements communication – Case one

railway. Those requirements are stored in a software called LIME, which is used to negotiate requirements between the client and the contractor. The documents stored in LIME are living documents, where both the client and supplier communicate the requirements and their feedback through an iterative process. After the client and the contractor have agreed on the requirements, those requirements are stored in DOORS[2].
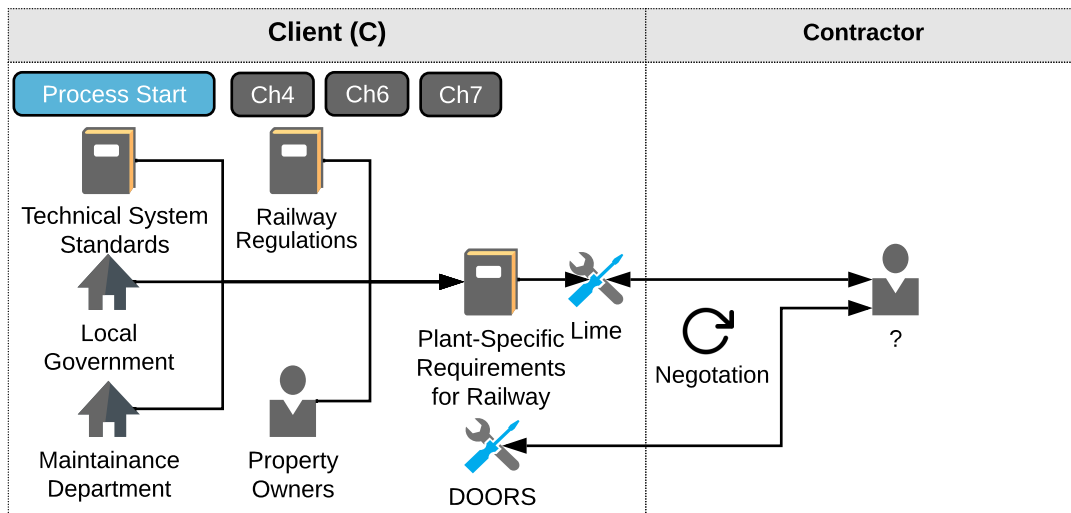

Figure 6. Requirements communication – Case two

The interviewees reported five challenges in relation to requirements communication.

4.2.1. Challenge 3: Misinterpretation of requirements

All parties working on the project have access to the same requirements documents, however, each party has their interpretation that could differ from the other parties' interpretation. This was reported in both cases by the design manager (A1), tender manager (A2), and discipline leader (B1). Moreover, the requirements can be unclear or difficult to understand. As one interviewee on the subcontractor side mentioned, "we have had a whole lot of discussions regarding these requirements in the technical description, and it is constantly not crystal clear, so it is very much up to interpretation" (B1). Also, the client thinks that

---

[2]Requirements management software.

they could improve their writing; as C4 said, "we often think that we are quite clear in our communication, but often it's not the case" (C4).

Lack of knowledge in local projects and *requirements are open for interpretation* are of the causes of this challenge, as seen by the interviewees. Some people working on the project may lack experience in projects from the this specific country, which could lead to a different interpretation of the requirements. "If you have any country A engineer and you want to make a bridge design, his background is slightly or completely different from the country B engineer" (A2). The subcontractor sees it as one of the causes of misinterpretation of requirements as put by one interviewee "There is never really a correct answer. There is not just one solution that you can do" (B1).

The misinterpretation of requirements may lead to rework and extra costs. If the subcontractor's interpretation is different from the client's interpretation, then additional work should be done by the subcontractor. As B1 said, "if our interpretation of the requirements is not that what the client wants us to do, then there is some additional work for us" (B1). In special cases, the misinterpretation of requirements could lead to extra costs paid by the contractor, as confirmed by A1, "I was not informed at all what a requirement (temporary use of land for the project) means".

4.2.2. Challenge 4: Long time to communicate requirements (changes and questions)

The requirements communication process between the different parties (client-supplier) takes a long time. This process includes requirements change requests and any requirements-related questions raised by the contractor or subcontractor. This challenge was seen in Case One and Case Two by the discipline leader (B1) and the head of environment (C4). All requirements-related communications between the client and subcontractor go through the contractor, which takes a long time: "it could take quite a bit of time" (B1). The client also sees that the change process takes a long time from the contractor side as well. C4 gave an example of a case, "one occasion we did a very big change of the requisition of the whole project ... but then it took like a year for the consultants to respond and tell us how the changes would be interpreted and implemented" (C4).

The interviewees did not explicitly identify the causes or consequences of this challenge.

4.2.3. Challenge 5: Finding the correct information

The documents communicated by the client do not include all the requirements that apply to the project, and it is challenging to find all requirements that apply by the contractor and subcontractor. This challenge was reported in Case One by the design manager (A4) and head of design (B2). As seen in Figure 5 and Figure 6, there are additional requirements documents that the client does not communicate as part of the tender documents. Those documents are local rules and regulations. The client only refers to those documents, and the contractor and subcontractor need to find those documents. "The tricky task to make sure you have all requirements that have to be followed" (A4).

A cause of this challenge is *Lack of knowledge in local projects*, similar to that of miss-interpretation of requirements challenge explained in Section 4.2.1. Some people with different background could lack experience in projects in a specific country. In this case, people with experience in local projects are consulted to find the right information.
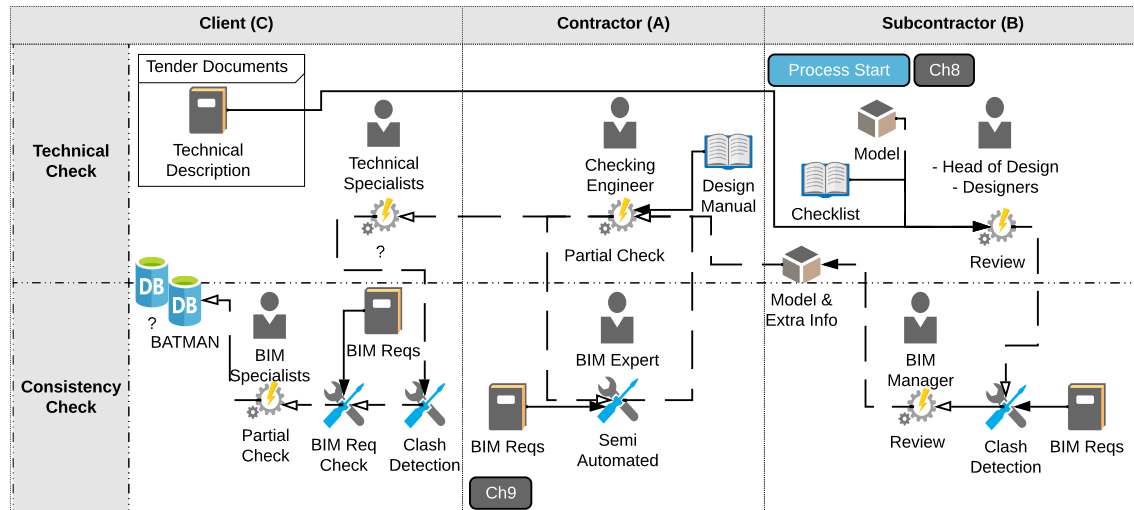
13

Figure 7. Digital assets verification – Case one

4.2.4. Challenge 6: Requirements elicitation and validation with non-technical stakeholders

The requirements elicitation and validation process is a challenging task since many stakeholders are involved. This challenge was reported in Case Two by a requirements engineer (C3). The client elicits requirements from many sources and stakeholders, as seen in Figure 6. Therefore, the way the client communicates the requirements needs to be adapted to the audience. "We can have a super model to communicate with the supplier, but we can't use that one when we meet the restaurant owner in the city" (C3).

The cause of seeing the requirements elicitation and validation process as a challenging task is because there are *different parties and many stakeholders with different backgrounds* involved in the process.

This challenge adds additional work to the client, as the client needs to adapt the requirements to the audience. For example, sketches and drawings need to be done so the client can communicate the requirements with the property owners.

4.2.5. Challenge 7: Requirements abstraction

The requirements communicated by the client would likely have variations in abstraction levels. The challenging part is to find the right level of granularity for these requirements. This is reported in Case Two by the head of environment (C4). A too specific requirement could constrain the supplier in developing the solution, "sometimes we were getting the wrong answer when we are too specific in demands" (C4), and a too abstract requirement is open for miss interpretation "often we have to explain more to make the requirements more specific" (C4).

We did not identify clear causes or consequences for this challenge from the interviews. The interviewee did not have a concrete answer about what could be causing such a challenge and what could be the consequences.

4.3. Digital assets verification

Figure 7 shows the digital assets verification process in Case One. There are two verification processes present in the figure. The technical check is the process of verifying the produced

14

digital assets' conformity with the specified technical requirements. The consistency check is the process of verifying the conformity of digital assets with the BIM requirements. BIM requirements specifies how the design models should be delivered (e.g., the file's extension to be used for delivering a model, or the units of measure used).

As the subcontractor designers produce the model, a review process is conducted by those designers and the head of design. The process takes as input 1) the produced model, which could be 2D drawings/3D models/BIM, 2) a checklist that is prepared by the designers based on the technical requirements before the implementation, 3) the technical requirements of the project. The output model of this process, along with BIM requirements, goes into a clash detection tool, which checks whether the models have any conflict in design objects. After that, a manual review process is made by the BIM manager.

At the contractor, a similar verification process on the model is followed. First, a technical check is done with the design manual, which is produced at the beginning of the project. This check is done by a checking engineer and is mainly based on experience with verification of similar models. The model is sent to the BIM expert who conducts pre-configured semi-automated checks on the attributes of the BIM objects based on the client's BIM requirements, using internal tools with customizable configuration possibilities. The technical and consistency check is an iterative process.

Then the model is sent to the client for approval. At the client, there are technical specialists who do their own technical checks while taking technical requirements as input. After that, a series of consistency checks are conducted: clash detection, BIM requirements check, and partial check by BIM specialists. When the model passes all those checks, it is stored in the appropriate database.
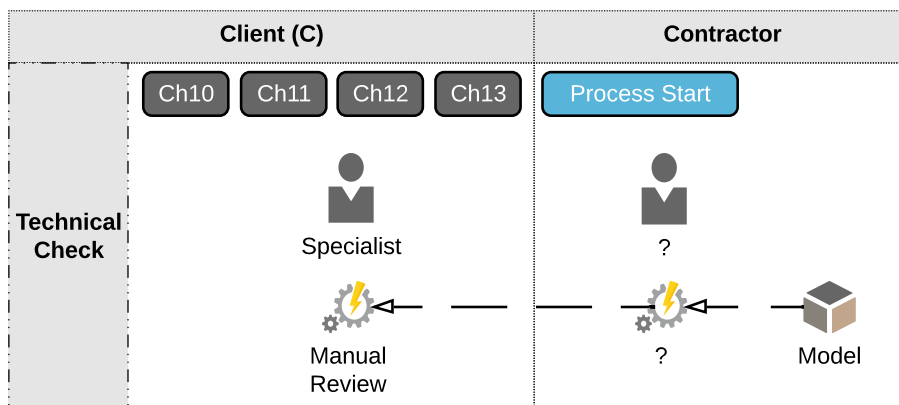

Figure 8. Digital assets verification – Case two

Figure 8 presents the digital assets verification process in Case Two. When the models are delivered to the client for approval, the specialist in the domain conducts a manual review process to verify those models. We also know that there is some kind of check being done by the contractor.

We have identified six challenges related to the process of digital assets verification.

4.3.1. Challenge 8: Requirements are impossible to build

In some cases, during the verification of digital assets, engineers detect requirements that are impossible to build. This challenge was reported in Case One by the discipline leader (B1). The contractor and subcontractor are limited by what they can change in the requirements.

Then during the verification of digital assets, some requirements appear to be impossible to build. "The solution that company C stated simply can't be done" (B1). Although the client may be validating the requirements, it is still the contractor's and subcontractor's responsibility to make sure those requirements are sound and possible to build.

The client specifies solutions rather than requirements were the cause of this challenge. The requirements provided by the client do not have much room for the contractor or subcontractor to come up with a solution. Rather the requirements specify an actual solution that the contractor and subcontractor need to follow. Therefore, if those requirements/solutions have conflicts, they will likely show during the verification process of the digital assets (since they might be overlooked in the requirements validation process, as discussed in Section 4.1.1.

The consequence of this challenge is *extra effort spent on rework* to be done by the subcontractor. During the verification process, if some requirements were detected to be impossible to build, then the subcontractor needs to come up with a new solution, verify it, and request a change for requirements.

4.3.2. Challenge 9: Difficulty understanding BIM requirements

Some consultants working on the project are not fluent or familiar with the language in which the requirements documents are written, as reported in Case One by the BIM manager (A3). It is difficult for those consultants to understand the BIM requirements. Although the consultants translate those documents into their language, they are not confident that the translation is accurate.

Consultants are unfamiliar with the documents' language was identified as the cause of this issue. "The language is always a problem when you are an international consultant" (A3).

Long time spent at the beginning is the consequence of this challenge. At the beginning of the project, it took a while for both the client and the contractor to coordinate and make sure that they were on the same page; as mentioned by A3 "It took some time coordinating with the client to understand the requirements" (A3).

This challenge is not specified to BIM requirements only. The project has many people (requirements engineers and designers) with different language proficiency, whom may find it difficult to interpret the requirements if the translation is inaccurate.

4.3.3. Challenge 10: Requirements management tool related

In Case Two, one of the identified challenges is related to using the requirements management tool DOORS. This challenge was reported by the requirements specialist (C2). People would prefer to use conventional tools like spreadsheets over a specialized new system like DOORS; as mentioned by C2, "specialists don't really like new systems" (C2). It is challenging to get people working on the tool. Another part of this challenge is related to reaching models from within DOORS. "When specialists review stuff, they need to go to the specific document in the specific models to look, and they can't just click on the link, which is annoying" (C2). The delivered models are verified in a different system, and it could be difficult to find the model and the related requirements during the review process by the client's engineers.

The causes we identified for this challenge are *use of yet another new tool* and *the limitation of the tool DOORS*. The requirements management tool DOORS is new for many

specialists working on the project, and they are not familiar with it; also, some specialists already maintain different systems. "It is a lot for people that already maintain like fifteen different systems so that was an issue to make them like DOORS" (C2). Also, DOORS has its limitations, e.g., "you can't do hyperlinks in the attribute in DOORS" (C2). Currently, using the tool to link the requirements to a specific attribute or part of the model is not possible.

This challenge leads to *unnecessary work for the requirements specialist* and *extra effort by the people verifying the models.* Since DOORS is a new system for people to use, the requirements specialist does spend time preparing views and arranging requirements to make it easy for people involved in the review process. "I have to be involved quite a bit just for them to know where to look" (C2). Additionally, since the tool has a limitation in linking requirements to the detailed model implementations, model verification requires extra effort. As explained by C2 "they need to be ready to open several different models which take time to load" (C2).

It is important to note here that this challenge does not point out the deficiencies of a specific tool, but rather the lack of integration of the many tools an engineer needs to use to perform their work.

4.3.4. Challenge 11: Granularity of traces

The traces created between the requirements and the models are of high abstraction. This was reported in Case Two by the requirements specialist (C2). There is information used in DOORS to trace requirements to the created models. However, this traceability information is not detailed enough, which makes it difficult to do the verification process. It is challenging to create those kinds of traces as it is seen to be an expensive practice.

Requirements are not linked to objects is seen to be as the cause of this challenge, "even though they give you a specific place to look there will be lots of places to look at" (C2). The current traces are created between requirements and models rather than requirements and the objects.

One consequence of this challenge is that an *extra effort is required to verify the model* as C2 explained "it just takes some time and effort" (C2). Since the requirements are linked to models rather than objects within the model, it takes extra effort and time from the specialist to verify the model.

4.3.5. Challenge 12: Lack of Experience Using Tools

There is a lack of experience in using the modeling tools by the people verifying the design models. This challenge was reported in Case Two by the requirements specialist (C2). The manual model review process, at the client in Case Two, depends mainly on the experience of the specialists. Although, DOORS contains traces between requirements and models, the specialists verifying the model must know where in the model the requirements apply. The specialists lack experience verifying those models, as (C2) put it "We are not very experienced using the model, so that's a challenge for all specialists" (C2).

The use of different models in the project is the cause of this challenge. There are too many model types used in the project; these models require different tools for viewing. It is difficult for specialists to cope with all these tools. C2 said, "we can not choose exactly what they are going to use; unfortunately, some consultants use different BIM modeling programs" (C2).

Difficulty navigating the models is a consequence of lacking the experience in verifying the models. Since the digital assets verification process at the client in Case Two is an experience-based process, it gets difficult to navigate the models to verify them. The specialist needs to figure out where in the model the specific requirements apply.

4.3.6. Challenge 13: Verifying all requirements

The client finds it challenging to verify all the requirements in the delivered model. This was reported in Case Two by the requirements specialist (C2). There are many generic and project-specific requirements that apply to the models; it becomes troublesome to verify whether the models conform to all requirements.

The *absence of risk analysis* for the requirements makes the verification of all requirements difficult. There exists no risk analysis nor classification of requirements based on severity or importance. Therefore, the client has no basis on which to base the prioritization for verification and resource allocation.

As a consequence, there is an *Uncertainty in the verification process* at the client side. Currently, it is not determined if a complete model verification, for all requirements that apply, is necessary or not. "There is a debate at the client whether or not we are supposed to do a complete verification or just sample verification and see" (C2).

5. Discussion

In this section, we discuss the implications of our study results for research and practitioners. We start by discussing the client-supplier communication cycle. Then we look at the challenges and identify potential solution areas from the literature. Finally, we discuss the role of the requirements documents in client-supplier communication.

5.1. Client-supplier communication

There are differences between the two cases in the way the requirements are communicated. In Case One, the client documented and released the requirements to the suppliers once, at the beginning of the project (Figure 5), while in Case Two, the requirements specification is a more mature process, as it adopts an iterative approach where the client and the contractor (i.e, supplier) negotiate the requirements before they are stored in a requirements management tool (Figure 6). Early requirements negotiation supports more correct and feasible requirements specification [47], and is part of the recipe of the system engineering best practices [48]. The requirements process in Case Two follows best practices and is, therefore, superior to the process in Case One.

The development of the requirements specifications in an infrastructure project is shared between the client and supplier, as the latter is expected to do their own analysis of the requirements, elicit the missing ones, and ensure that they align with the client's expectations. This is in contrast to software engineering projects where the client is normally responsible for sharing a set of requirements that aligns with their needs. However, in the public domain, the development of the requirements specifications in software projects could be similar to infrastructure projects. In a case study conducted by Brataas et al., [49] to investigate an innovation partnership project in the Norwegian medical sector, the development of the requirements was a joint effort between the client procurement team

and multiple suppliers, which allowed the procurement team to focus on innovation. Moe et al. [50] made a similar observation in their study on information systems requirements in public sector projects: the dialogue between the client and supplier complements the project's requirements specification.

In Case One, every party involved in the project does their verification for the delivered digital assets. The technical check, as we see in Figures 7 and 8, is a sample check done manually. There are two drawbacks of this check: 1) it is time consuming, and 2) it does not verify all parts of the delivered digital assets. The supplier takes responsibility for the issues that show later in the project due to unverified requirements. A similar observation was made by Makkinga et al. [19], where a supplier carried the responsibility of the verification when the contractor did not have enough resources to conduct a complete verification. Consistency checks, which verify whether the design conforms to the general design requirements such as correct use of units of measurement or file formats, are done (semi-)automatically, as shown in Figure 7. We believe that developing a common checklist would have been beneficial in the early identification of potential issues to requirements. Moreover, we speculate that communication between two parties with various levels of domain knowledge could be facilitated by communication brokers [51].

The life cycle model used in both cases is sequential. A sequential model is defined by INCOSE [52] as a systematic approach for the system development process where the system goes through a sequence of steps from goals definition to a complete system. This sequential model is beneficial to use in our cases which are large projects with different parties involved. However, this model also requires verified traceable requirements [52], which is challenging to achieve. As we saw in Section 4, the requirements validation may be overlooked (ch3), and the traceability information is not adequate (ch11), which makes checking all the requirements difficult (ch13).

5.2. Challenges and potential solutions

The mapping of challenges to solutions was done based on the following: (1) Analysis and synthesis of the results, as some possible solutions were identified during the interview process, e.g., feedback during RE process, policy change and tender process. (2) The authors' experience in requirements engineering research

Figure 9[3] depicts the identified challenges, their causes, and their consequences and shows the relations between them based on the analysis of the study results and complemented with references from the literature based on the author's experience in requirements engineering research. In addition, we propose solutions areas to tackle these challenges. In the problem analysis, arrows represent a *contributes to* relation between the connected elements. The arrows from the solution area to the problem area indicate whether a solution *potentially addresses* the connected cause and consequently the connected challenge(s). We differentiate between whether the relation originates from:
– our results (solid arrow)
– an observation made in another study (dashed arrow with reference)
– both from our results and from literature (solid arrow with reference)

The solution areas proposed in this section are not exhaustive for all possible solutions for the challenges and their causes. Rather, they are potential solutions where their effectiveness in addressing the challenges has been reported in literature but still needs to

---

[3]We will make the citations in the figure consistent with the citation in the paper before publishing.
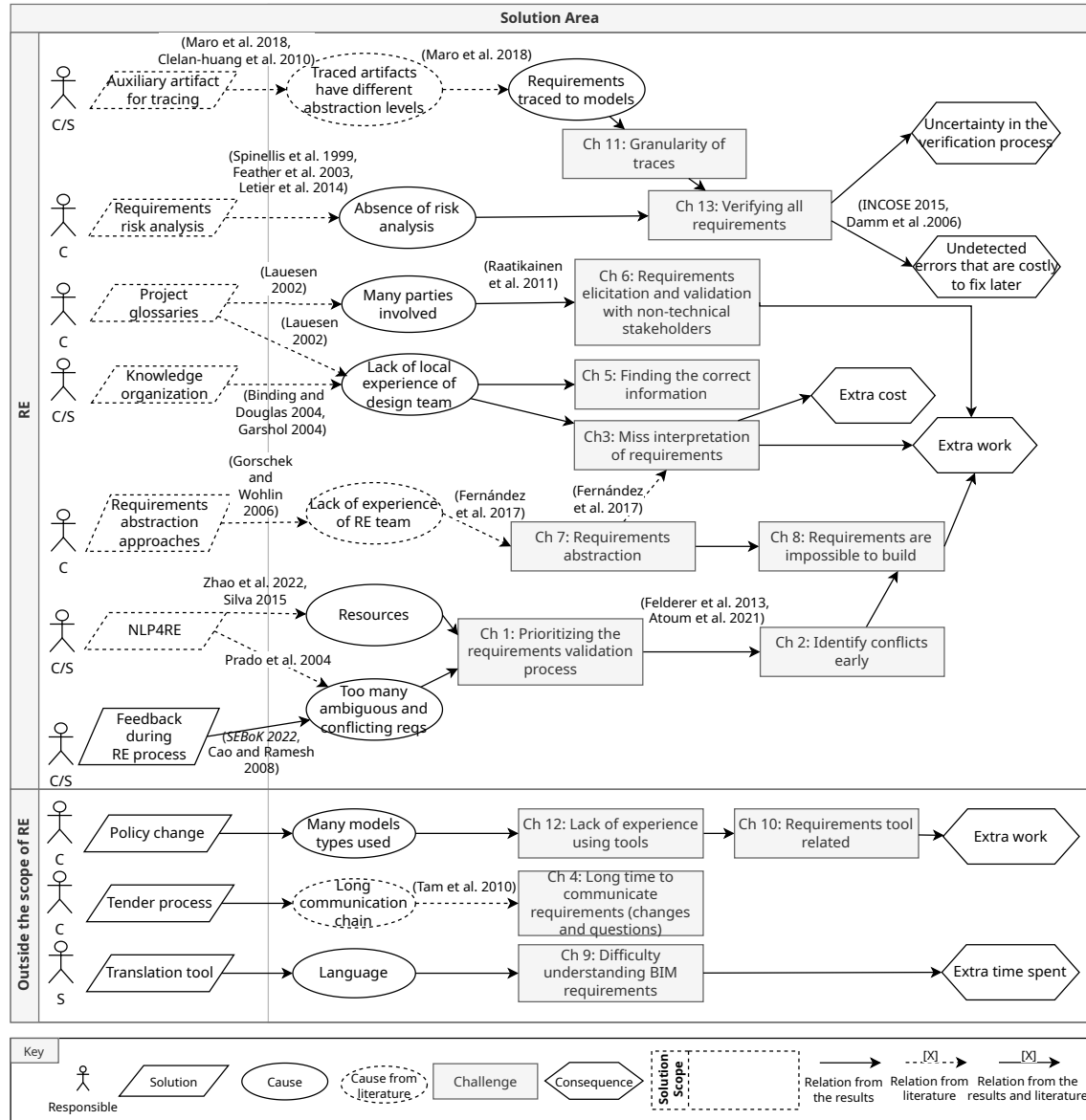
Figure 9. Relationship between challenges, the abbreviation (ch) refers to challenge

be verified in the particular context we have studied. Furthermore, we elaborate on where a solution could be implemented, at the client or the supplier side, by indicating the main *responsible party*. We present the solution areas in more detail next.

Auxiliary artifacts in tracing When tracing requirements to downstream artifacts in software (e.g., test cases) and system (e.g., design models) projects, an auxiliary artifact could be used to address the different abstraction levels of the traced artifacts. Thus, the auxiliary artifact should be traceable to artifacts with different abstraction levels. It can be an artifact that is produced during the project execution, e.g., a lower level requirement [57], or an external artifact that is not part of the software or system, e.g., a domain ontology containing domain concepts and has a hierarchical structure [58]. Using an auxiliary artifact solves the abstraction level mismatch between requirements and the downstream artifacts (e.g., test cases or design models), and consequently could address the granularity of traces challenge (ch11). Fine-grained trace links make it possible to trace requirements to their

implementation; thus, verifying all requirements (ch13) in the design model can be more feasible. This solution should be implemented by the client (for the requirements) and by the supplier (for the delivered artifact).

Requirements risk analysis A risk analysis could be conducted on all requirements to assess the severity and consequences of failing to fulfill a requirement. A requirements-risk analysis is usually conducted in a similar way as those done for a project plan. In the requirements-risk analysis, three main concepts are identified: problematic requirements, the possible failures, and activities that could prevent or mitigate these risks [59]. The risk analysis results help engineers in decision-making by reducing uncertainty [60]. Such decisions are where resources should be spent [59] or which requirements should be prioritized for verification [61]. The client should be responsible for introducing requirements risk analysis as part of the RE process to ensure that the project adheres to the requirements that threaten the project success.

Project glossaries Project glossaries consist of a set of terms and their descriptions in a specific domain or a project. Project glossaries are used to mitigate misinterpretations of terms (ch3) by project members with different backgrounds [62]. Implementing project glossaries can address differences in experience of design teams align terminology between parties with different backgrounds involved during requirements elicitation (ch6).

Project glossaries address these issues by providing an explicit description of terms that can be misunderstood. Delisle and Olson investigate whether project-based terminology and definitions are actually as widely accepted as believed and conclude that more effort should be dedicated towards coordination of glossaries and dissemination of information about project management terms and definitions [63]. We speculate that a broader adoption of glossaries can be supported by automated construction of project glossaries [64, 65]. In our case, the client specifies the project requirements, so it is his responsibility to develop and maintain the project glossary.

Knowledge organization Knowledge organization is the process of indexing, classifying, and archiving documents and books to ease access to them [66]. McClory et al. stressed the importance of knowledge management and organisational learning and suggested triple-loop learning as an organisational structure [67].

Infrastructure projects contain many documents (e.g., rules and regulations) that need to be structured and organized to be useful for requirements extraction and analysis. Garshol [68] identified different types of knowledge organization systems, e.g., controlled vocabularies and taxonomies. The use of an appropriate knowledge organization system to structure domain knowledge [69] could allow engineers to find information about local projects more efficiently and effectively (ch5). This solution could be introduced either by the client or the supplier. The client owns the rules and regulations, and organizing them through automated classification makes access to information more efficient. The supplier could be obligated to adhere to international standards which apply to multiple projects.

Requirements abstraction methods First, one of the causes of the major challenges in requirements engineering is the existence of too abstract requirements, which could result in miss interpretation (ch3) [70]. Thus, a requirement should not be too abstract or unambiguous [71]. Second, according to SWEBOK, a requirement is defined as "a property that must be exhibited by something in order to solve some problem in the real world" and a good SRS should be an agreement between the client and supplier about "what the software product is to do", not how to do it [71]. Thus, a requirement should not restrict possible solutions or be impossible to be realized (ch8). We argue that

requirement abstraction methods should be applied to ensure requirements are written on an appropriate abstraction level. Gorschek and Wohlin [72] have developed the Requirements Abstraction Model (RAM), a method to systematically specify requirements on multiple levels of abstraction (from product to component level). RAM has been shown to improve the requirements engineering process and the quality of the requirements specified in practice [73]. Liebel et al. suggested coordinating requirements on various abstraction to avoid communication and coordination breakpoints [30]. Requirements engineers on the client side should adopt one of these solutions as part of the RE process.

NLP4RE Natural language processing (NLP) is the use of (semi-)automated techniques to analyze and model human language [74], mainly through employing machine learning. This is particularly beneficial as the majority of requirements in software and system projects are specified in natural language [75, 76]. NLP approaches have shown their potential to support the requirements validation process as presented by Zhao et al. [77] in their review of the literature on NLP for requirements engineering (NLP4RE). The use of NLP4RE reduces the number of resources required for the requirements validation [77, 78], e.g., by performing automated model checking. Moreover, NLP4RE supports requirements engineers in identifying conflicts and ambiguities (ch2), which requires expertise [79]. Both the client and supplier could benefit from adopting this solution in the RE process.

Feedback during RE process One of the good practices in system engineering [48] and software engineering [80] during the requirements engineering process is adopting an iterative approach, where the client and supplier agree on the requirements to be developed incrementally. This approach increases the understanding of requirements and helps with their prioritization [81]. In Case Two, an iterative approach for RE is adopted where the client and the contractor (i.e., supplier) negotiate the requirements before they are stored in a requirements management tool (Figure 6). However, in Case One, the requirements were specified and communicated up-front, and changes were difficult to introduce due to contractual obligations. Adopting an iterative approach when specifying the requirements where the feedback of the supplier is considered reduces ambiguity and conflict in the specified requirements. Both the client and the supplier are responsible for implementing this solution.

Outside the scope of RE Other challenges (4,9,10,12) and solutions areas are outside the scope of RE but in the area of project management. A change in companies' policy (client), an improved tender process (client), and using better tools are required in order to tackle these challenges.

5.3. Impact on Academia and Industry

This study illustrates the persistence of challenges related to requirements. Since existing solutions may not be effective, a root-cause analysis should be conducted to identify the causes behind these challenges and find solutions to address them. Furthermore, the requirements process and identified challenges in infrastructure projects are similar to those in the software engineering domain. Thus, although they are two distinct domains, they share many similarities when it comes to requirement engineering. The software and system engineering domain researchers should learn from each other's experiences in investigating requirements engineering processes and addressing challenges related to them.

When researchers and practitioners are looking for a solution to specific challenges in one domain, they should explore existing solutions in the other. The challenges that we

identified in this study and their mapping to potential solutions, as we presented in this section, illustrate the similarity between the system and software engineering domain. Both domains use natural language to specify requirements, and both have digital assets that need to be verified against the specified requirements. In the software domain, engineers produce digital artifacts such as design documents and source code, while in the system domain, engineers produce design models (digital twins of the system) before building the physical build. Moreover, adopting requirements communication techniques is highly important for the system domain in general and the infrastructure domain in particular, due to the sub-contracted work being the norm, which adds more complexity to the requirements communication process.

5.4. Threats to validity

We use the framework by Runesson et al. [46] to discuss the validity threats of our study. The framework lists four categories of case study validity: construct, internal, external, and reliability. We analyzed the validity threats of our study from the beginning of our study, and we continued to revise these threats to minimize them.

Construct Validity We mitigated threats to design and execution by involving more than one researcher in designing and conducting the study. The involved researchers reviewed the research questions and the case study protocol. Feedback sessions were conducted to discuss the protocol and make improvements. For example, the first author wrote a set of interview questions per studied process. During a feedback session, the third author suggested that the questions should be adapted to each interviewee's role. On that premise, the interview questions were adapted to each role by the first author, and the other researchers reviewed them. The improvement of the interview questions for each interview was done by the researchers before the start of said interview.

Internal Validity Although the involved researchers revised and improved the case study protocol, there might be a bias or errors in the collected data. To reduce the risk of bias in data collection, between two and three researchers were present during each interview. One researcher led the interview, and the other researchers observed the interview and asked follow-up questions if necessary. In addition, after an interview was completed, we transcribed the interview, a second researcher verified the transcript and finally sent it out to the interviewee for comments.

Another internal threat to validity is the risk of bias when coding the transcripts as part of the data analysis phase. We mitigate this threat by assigning the coding task to one researcher and the verification of those codes to another. Eventual disagreements were resolved in meetings.

External Validity It could be argued that generalization would be difficult with two case studies and that more case studies may be required to increase the validity of the study and achieve generalization. However, generalization can be achieved by individual cases. It is a matter of the selected cases. As argued by Flyvbjerg, one of the five misunderstandings of a case study is the inability to generalize from a single case [82]. We focus on analytical generalization rather than statistical generalization by providing detailed case descriptions and discussing the implications of our findings. Moreover, we believe that our cases are a good representation of the population (infrastructure projects). They are two large infrastructure projects, with large-sized (45,000+ employees) companies involved. Case One is in its final stages, while Case Two is in its early stages.

Reliability To ensure that our study is repeatable, we present in Section 3 the protocol of our study. We explain the setup of our studied cases, list the cluster of the interview questions, present the steps in which we conducted the interviews, and the coding mechanism of the interview data.

6. Conclusion

We have conducted two case studies in three companies to explore the requirements validation, requirements communication, digital assets verification processes, and the challenges associated with these processes between the client and the supplier in infrastructure projects. We identified 13 challenges and proposed potential solutions from the literature to address these challenges. Many of the challenges faced, during the forward communication (tender documents) and backward communication (project deliverables), between the client and supplier can be addressed in the area of requirements engineering. Furthermore, the solution for similar challenges in the software domain can potentially address the challenges observed in the system domain.

The system requirements play a main role in the communication between the client and supplier in infrastructure projects, and their quality likely affects subsequent processes, e.g., verification and acceptance. The verification process is a difficult task in projects on this large scale, with thousands of requirements and models to validate. One particular difficulty is keeping track of the requirements during the verification process.

7. Future work

We have identified several future work directions based on the finding from this paper, that we outline below:
– Exploring methods to introduce early requirements risk analysis and how to estimate potential risks on early and often incomplete requirements and how to enable risk-based requirements reasoning [59].
– Exploring auxiliary artifacts and their usage scenario in requirements traceability. This involves exploring the relationship between the structural characteristics of auxiliary artifacts (e.g., a taxonomy) and the performance of machine learning models for requirements traceability.
– Investigating what types of knowledge organization systems are the most suitable for large-scale infrastructure projects – we believe that project glossaries should be considered as an efficient way of organizing knowledge and building common vocabulary between the parties. We also plan to explore the suitability of topic maps knowledge organizational structure for infrastructure projects [68].
– Investigating the suitability of requirements abstraction models from the software domain (e.g., RAM [73]) for the infrastructure projects.
– Exploring the use of NLP4RE techniques to detect conflicts and ambiguity in functional requirements from the infrastructure domain.

CRediT authorship contribution statement

Waleed Abdeen: conceptualization, methodology, formal analysis, investigation, and writing
– original draft. Krzysztof Wnuk: conceptualization, methodology, investigation, supervision,
and writing – review & editing. Michael Unterkalmsteiner: conceptualization, methodology,
formal analysis and investigation, supervision, project administration, funding acquisition,
and writing – review & editing. Alexandros Chirtoglou: conceptualization, formal analysis
and writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal
relationships that could have appeared to influence the work reported in this paper.

Funding

References

[1] H. Hofmann and F. Lehner, "Requirements engineering as a success factor in software projects,"
IEEE Software, Vol. 18, No. 4, Jul. 2001, pp. 58–66.

[2] E. Bjarnason, K. Wnuk, and B. Regnell, "Requirements are slipping through the gaps—a case
study on causes & effects of communication gaps in large-scale software development," in *2011
IEEE 19th international requirements engineering conference*. IEEE, 2011, pp. 37–46.

[3] O. Liskin, "How artifacts support and impede requirements communication," in *Requirements
Engineering: Foundation for Software Quality: 21st International Working Conference, REFSQ
2015, Essen, Germany, March 23-26, 2015. Proceedings 21*. Springer, 2015, pp. 132–147.

[4] P. Brereton, "The software customer/supplier relationship," *Commun. ACM*, Vol. 47, No. 2,
Feb. 2004, p. 77–81. [Online]. https://doi.org/10.1145/966389.966394

[5] C. Ebert, "Optimizing supplier management in global software engineering," in *International
Conference on Global Software Engineering (ICGSE 2007)*, 2007, pp. 177–185.

[6] T. Vullinghs, T. Gantner, S. Steinhauer, and T. Weber, "Experiences on lean techniques
to manage software suppliers," in *Product Focused Software Process Improvement: Second
International Conference, PROFES 2000, Oulu, Finland, June 20-22, 2000. Proceedings 2*.
Springer, 2000, pp. 244–256.

[7] Y. Zheng, S. Liu, W. Huang, and J.J.Y. Jiang, "Inter-organizational cooperation in automotive
new product development projects," *Industrial Management & Data Systems*, Vol. 120, No. 1,
2020, pp. 79–97.

[8] J.J. Jiang, G. Klein, J.C.A. Tsai, and Y. Li, "Managing multiple-supplier project teams in new
software development," *International Journal of Project Management*, Vol. 36, No. 7, 2018,
pp. 925–939. [Online]. https://www.sciencedirect.com/science/article/pii/S026378631830125X

[9] J.S. Aurangzeab Butt, Marja Naaranoja, "Project change stakeholder communication," *Interna-
tional Journal of Project Management*, Vol. 34, No. 8, 2016, pp. 1579–1595.

[10] G. Wu, C. Liu, X. Zhao, and J. Zuo, "Investigating the relationship between
communication-conflict interaction and project success among construction project teams,"
International Journal of Project Management, Vol. 35, No. 8, 2017, pp. 1466–1482.

[11] Y.F. Badir, B. Büchel, and C.L. Tucci, "A conceptual framework of the impact of npd project team and leader empowerment on communication and performance: An alliance case context," *International Journal of Project Management*, Vol. 30, No. 8, 2012, pp. 914–926.

[12] H.Ç. Demirel, W. Leendertse, L. Volker, and M. Hertogh, "Flexibility in ppp contracts–dealing with potential change in the pre-contract phase of a construction project," *Construction management and economics*, Vol. 35, No. 4, 2017, pp. 196–206.

[13] A.D. Songer and K.R. Molenaar, "Project characteristics for successful public-sector design-build," *Journal of construction engineering and management*, Vol. 123, No. 1, 1997, pp. 34–40.

[14] C.W. Ibbs, Y.H. Kwak, T. Ng, and A.M. Odabasi, "Project delivery systems and project change: Quantitative analysis," *Journal of construction engineering and management*, Vol. 129, No. 4, 2003, pp. 382–387.

[15] E. Kania, E. Radziszewska-Zielina, and G. Śladowski, "Communication and information flow in polish construction projects," *Sustainability*, Vol. 12, No. 21, 2020, p. 9182.

[16] Z.Y. Zhao, Q.L. Lv, J. Zuo, and G. Zillante, "Prediction system for change management in construction project," *Journal of Construction Engineering and Management*, Vol. 136, No. 6, 2010, pp. 659–669.

[17] S. Malik, M. Taqi, J.M. Martins, M.N. Mata, J.M. Pereira et al., "Exploring the relationship between communication and success of construction projects: The mediating role of conflict," *Sustainability*, Vol. 13, No. 8, 2021, p. 4513.

[18] S.D. Vermillion and R.J. Malak, "An investigation on requirement and objective allocation strategies using a principal–agent model," *Systems Engineering*, Vol. 23, No. 1, 2020, pp. 100–117.

[19] R. Makkinga, R.d. Graaf, and H. Voordijk, "Successful verification of subcontracted work in the construction industry," *Systems Engineering*, Vol. 21, No. 2, 2018, pp. 131–140.

[20] A.M. Madni and M. Sievers, "Systems integration: Key perspectives, experiences, and challenges," *Systems Engineering*, Vol. 17, No. 1, 2014, pp. 37–51.

[21] J. Xu, H. Smyth, and V. Zerjav, "Towards the dynamics of trust in the relationship between project-based firms and suppliers," *International Journal of Project Management*, Vol. 39, No. 1, 2021, pp. 32–44.

[22] L.S. Henderson, "Encoding and decoding communication competencies in project management – an exploratory study," *International Journal of Project Management*, Vol. 22, No. 6, 2004, pp. 469–476.

[23] M. Loosemore and H. Muslmani, "Construction project management in the persian gulf: inter-cultural communication," *International Journal of Project Management*, Vol. 17, No. 2, 1999, pp. 95–100.

[24] C. Semeraro, M. Lezoche, H. Panetto, and M. Dassisti, "Digital twin paradigm: A systematic literature review," *Computers in Industry*, Vol. 130, 2021, p. 103469.

[25] A. Toygar, "A new asset type: Digital assets," *Journal of International Technology and Information Management*, Vol. 22, No. 4, 2013, p. 9.

[26] P.E.D. Love, J. Zhou, J. Matthews, and H. Luo, "Systems information modelling: Enabling digital asset management," *Advances in Engineering Software*, Vol. 102, 2016, pp. 155–165.

[27] A.R. Hevner, S.T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS quarterly*, 2004, pp. 75–105.

[28] R.J. Wieringa, *Design science methodology for information systems and software engineering.* Springer, 2014.

[29] E. Bjarnason, B. Gislason Bern, and L. Svedberg, "Inter-team communication in large-scale co-located software engineering: a case study," *Empirical Software Engineering*, Vol. 27, No. 2, 2022, p. 36.

[30] G. Liebel, M. Tichy, E. Knauss, O. Ljungkrantz, and G. Stieglbauer, "Organisation and communication problems in automotive requirements engineering," *Requirements Engineering*, Vol. 23, 2018, pp. 145–167.

[31] J. Iqbal, R. Ahmad, M.H.N.B.M. Nasir, and M.A. Noor, "A framework to address communication issues during requirements engineering process for software development outsourcing," *Journal of Internet Technology*, Vol. 19, No. 3, 2018, pp. 845–859.

[32] S. Fricker, T. Gorschek, C. Byman, and A. Schmidle, "Handshaking: negotiate to provoke the right understanding of requirements," *IEEE Software*, Vol. 99, 2009.

[33] J. Pernstål, T. Gorschek, R. Feldt, and D. Florén, "Requirements communication and balancing in large-scale software-intensive product development," *Information and Software Technology*, Vol. 67, 2015, pp. 44–64.

[34] S. Maalem and N. Zarour, "Challenge of validation in requirements engineering," *Journal of Innovation in Digital Ecosystems*, Vol. 3, No. 1, 2016, pp. 15–21.

[35] T.F. Lynghaug, S. Kokkula, and G. Muller, "Investigating systems engineering approaches in the norwegian construction industry: A multi-case study," *INCOSE International Symposium*, Vol. 32, No. 1, 2022, pp. 792–808.

[36] R. de Graaf, R. Pater, and H. Voordijk, "Level of sub-contracting design responsibilities in design and construct civil engineering bridge projects," *Frontiers in Engineering and Built Environment*, Vol. 3, No. 3, 2023, pp. 192–205, publisher: Emerald Publishing Limited.

[37] M. Raatikainen, T. Männistö, T. Tommila, and J. Valkonen, "Challenges of requirements engineering—a case study in nuclear energy domain," in *2011 IEEE 19th International Requirements Engineering Conference*. IEEE, 2011, pp. 253–258.

[38] D. Saxena and J. McDonagh, "Communication breakdowns during business process change projects – insights from a sociotechnical case study," *International Journal of Project Management*, Vol. 40, No. 3, 2022, pp. 181–191.

[39] E. Ochieng and A. Price, "Managing cross-cultural communication in multicultural construction project teams: The case of kenya and uk," *International Journal of Project Management*, Vol. 28, No. 5, 2010, pp. 449–460.

[40] T.U. Daim, A. Ha, S. Reutiman, B. Hughes, U. Pathak et al., "Exploring the communication breakdown in global virtual teams," *International Journal of Project Management*, Vol. 30, No. 2, 2012, pp. 199–212.

[41] L.S. Henderson, R.W. Stackman, and R. Lindekilde, "The centrality of communication norm alignment, role clarity, and trust in global project teams," *International Journal of Project Management*, Vol. 34, No. 8, 2016, pp. 1717–1730.

[42] A. Terry Bahill and S.J. Henderson, "Requirements development, verification, and validation exhibited in famous failures," *Systems engineering*, Vol. 8, No. 1, 2005, pp. 1–14.

[43] E. Bjarnason, P. Runeson, M. Borg, M. Unterkalmsteiner, E. Engström et al., "Challenges and practices in aligning requirements with verification and validation: a case study of six companies," *Empirical software engineering*, Vol. 19, No. 6, 2014, pp. 1809–1855.

[44] S. Hotomski, E.B. Charrada, and M. Glinz, "An exploratory study on handling requirements and acceptance test documentation in industry," in *2016 IEEE 24th International Requirements Engineering Conference (RE)*, 2016, pp. 116–125, ISSN: 2332-6441.

[45] A.R. Hevner, "A three cycle view of design science research," *Scandinavian journal of information systems*, Vol. 19, No. 2, 2007, p. 4.

[46] P. Runeson and M. Höst, "Guidelines for conducting and reporting case study research in software engineering," *Empirical software engineering*, Vol. 14, No. 2, 2009, p. 131.

[47] P. Grünbacher and N. Seyff, "Requirements Negotiation," in *Engineering and Managing Software Requirements*, A. Aurum and C. Wohlin, Eds. Berlin, Heidelberg: Springer, 2005, pp. 143–162.

[48] SEBoK, "Stakeholder needs and requirements — sebok‚" 2020, [Online; accessed 15-February-2021].

[49] G. Brataas, G.K. Hanssen, X. Qiu, and L.S. Græslie, "Requirements engineering in the market dialogue phase of public procurement: A case study of an innovation partnership for medical technology," in *Requirements Engineering: Foundation for Software Quality*, V. Gervasi and A. Vogelsang, Eds. Cham: Springer International Publishing, 2022, pp. 159–174.

[50] C.E. Moe, M. Newman, and M.K. Sein, "The public procurement of information systems: dialectics in requirements specification," *European Journal of Information Systems*, Vol. 26, No. 2, 2017, pp. 143–163.

[51] D. Damian, R. Helms, I. Kwan, S. Marczak, and B. Koelewijn, "The role of domain knowledge and cross-functional communication in socio-technical coordination," in *2013 35th International Conference on Software Engineering (ICSE)*. IEEE, 2013, pp. 442–451.

[52] INCOSE and Wiley, *INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*. New York: John Wiley & Sons, Incorporated, 2015.

[53] L.O. Damm, L. Lundberg, and C. Wohlin, "Faults-slip-through—a concept for measuring the efficiency of the test process," *Software Process: Improvement and Practice*, Vol. 11, No. 1, 2006, pp. 47–59.

[54] I. Atoum, M.K. Baklizi, I. Alsmadi, A.A. Otoom, T. Alhersh et al., "Challenges of software requirements quality assurance and validation: A systematic literature review," *IEEE Access*, Vol. 9, 2021, pp. 137 613–137 634.

[55] M. Felderer and A. Beer, "Using defect taxonomies for requirements validation in industrial projects," in *2013 21st IEEE International Requirements Engineering Conference (RE)*. IEEE, 2013, pp. 296–301.

[56] V.W.Y. Tam, L.Y. Shen, and J.S.Y. Kong, "Impacts of multi-layer chain subcontracting on project management performance," *International Journal of Project Management*, Vol. 29, No. 1, 2011, pp. 108–116.

[57] S. Maro, J.P. Steghöfer, and M. Staron, "Software traceability in the automotive domain: Challenges and solutions," *Journal of Systems and Software*, Vol. 141, 2018, pp. 85–110.

[58] J. Cleland-Huang, A. Czauderna, M. Gibiec, and J. Emenecker, "A machine learning approach for tracing regulatory codes to product specific requirements," in *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 1*, ICSE '10. Association for Computing Machinery, 2010, pp. 155–164.

[59] M.S. Feather and S.L. Cornford, "Quantitative risk-based requirements reasoning," *Requirements Engineering*, Vol. 8, No. 4, 2003, pp. 248–265.

[60] E. Letier, D. Stefan, and E.T. Barr, "Uncertainty, risk, and information value in software requirements and architecture," in *Proceedings of the 36th International Conference on Software Engineering*, ICSE 2014. Association for Computing Machinery, 2014, pp. 883–894.

[61] D. Spinellis, S. Kokolakis, and S. Gritzalis, "Security requirements, risks and recommendations for small enterprise and home-office environments," *Information Management & Computer Security*, Vol. 7, No. 3, 1999, pp. 121–128, publisher: MCB UP Ltd.

[62] S. Lauesen, *Software requirements: styles and techniques*. Pearson Education, 2002.

[63] C.L. Delisle and D. Olson, "Would the real project management language please stand up?" *International Journal of Project Management*, Vol. 22, No. 4, 2004, pp. 327–337.

[64] C. Arora, M. Sabetzadeh, L. Briand, and F. Zimmer, "Automated extraction and clustering of requirements glossary terms," *IEEE Transactions on Software Engineering*, Vol. 43, No. 10, 2017, pp. 918–945.

[65] C. Wang, X. Peng, M. Liu, Z. Xing, X. Bai et al., "A learning-based approach for automatic construction of domain glossary from source code and documentation," in *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ESEC/FSE 2019. Association for Computing Machinery, 2019, pp. 97–108.

[66] B. Hjørland, "What is knowledge organization (ko)?" *KO Knowledge Organization*, Vol. 35, No. 2-3, 2008, pp. 86–101.

[67] S. McClory, M. Read, and A. Labib, "Conceptualising the lessons-learned process in project management: Towards a triple-loop learning framework," *International Journal of Project Management*, Vol. 35, No. 7, 2017, pp. 1322–1335, social Responsibilities for the Management of Megaprojects.

[68] L.M. Garshol, "Metadata? thesauri? taxonomies? topic maps! making sense of it all," *Journal of Information Science*, Vol. 30, No. 4, 2004, pp. 378–391, publisher: SAGE Publications Ltd.

[69] C. Binding and D. Tudhope, "Kos at your service: programmatic access to knowledge organisation systems," *Journal of Digital Information*, 2004.

[70] D.M. Fernández, S. Wagner, M. Kalinowski, M. Felderer, P. Mafra et al., "Naming the pain in requirements engineering: Contemporary problems, causes, and effects in practice," *Empirical software engineering*, Vol. 22, 2017, pp. 2298–2338.

[71] IEEE, "IEEE recommended practice for software requirements specifications," *IEEE Std 830-1998*, 1998, pp. 1–40.

[72] T. Gorschek and C. Wohlin, "Requirements abstraction model," *Requirements Engineering*, Vol. 11, No. 1, 2006, pp. 79–101.

[73] T. Gorschek, P. Garre, S.B.M. Larsson, and C. Wohlin, "Industry evaluation of the requirements abstraction model," *Requirements Engineering*, Vol. 12, No. 3, 2007, pp. 163–190.

[74] J. Hirschberg and C.D. Manning, "Advances in natural language processing," *Science*, Vol. 349, No. 6245, 2015, pp. 261–266, publisher: American Association for the Advancement of Science.

[75] M. Kassab, C. Neill, and P. Laplante, "State of practice in requirements engineering: contemporary data," *Innovations in Systems and Software Engineering*, Vol. 10, No. 4, 2014, pp. 235–241.

[76] S. Wagner, D.M. Fernández, M. Felderer, A. Vetrò, M. Kalinowski et al., "Status quo in requirements engineering: A theory and a global family of surveys," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, Vol. 28, No. 2, 2019, pp. 1–48.

[77] L. Zhao, W. Alhoshan, A. Ferrari, K.J. Letsholo, M.A. Ajagbe et al., "Natural language processing for requirements engineering: A systematic mapping study," *ACM Computing Surveys*, Vol. 54, No. 3, 2022, pp. 1–41.

[78] A.R. da Silva, "SpecQua: Towards a framework for requirements specifications with increased quality," in *Enterprise Information Systems*, Lecture Notes in Business Information Processing, J. Cordeiro, S. Hammoudi, L. Maciaszek, O. Camp, and J. Filipe, Eds. Springer International Publishing, 2015, pp. 265–281.

[79] F.A.C. Pinheiro, *Perspectives on Software Requirements*. Boston, MA: Springer US, 2004, Ch. Requirements Traceability, pp. 91–113.

[80] B. Ramesh, L. Cao, and R. Baskerville, "Agile requirements engineering practices and challenges: an empirical study," *Information Systems Journal*, Vol. 20, No. 5, 2010, pp. 449–480.

[81] L. Cao and B. Ramesh, "Agile requirements engineering practices: An empirical study," *IEEE Software*, Vol. 25, No. 1, 2008, pp. 60–67.

[82] B. Flyvbjerg, "Five misunderstandings about case-study research," *Qualitative Inquiry*, Vol. 12, No. 2, 2006, pp. 219–245, publisher: SAGE Publications Inc.

Authors and affiliations

Waleed Abdeen
e-mail: waleed.abdeen@bth.se
ORCID: https://orcid.org/0000-0001-8142-9631
Software Engineering, BTH, Sweden

Krzysztof Wnuk
e-mail: krzysztof.wnuk@bth.se
ORCID: https://orcid.org/0000-0003-3567-9300
Software Engineering, BTH, Sweden

Michael Unterkalmsteiner
e-mail: michael.unterkalmsteiner@bth.se
ORCID: https://orcid.org/0000-0003-4118-0952
Software Engineering, BTH, Sweden

Alexandros Chirtoglou
e-mail: alexandros.chirtoglou@hochtief.de
HOCHTIEF ViCon GmbH, Germany