CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG

Challenges of Integrating Diverse Requirements Documentation and Traceability Practices: A case study in large-scale agile systems development

Master's thesis in Computer science and engineering

Kiana Ghanbari Adib

Atiye Etemadi

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2022

Master's thesis 2022

Challenges of Integrating Diverse Requirements Documentation and Traceability Practices: A case study in large-scale agile systems development

Kiana Ghanbari Adib

Atiye Etemadi

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2022

Challenges of Integrating Diverse Requirements Documentation and Traceability Practices: A case study in large-scale agile systems development

Kiana Ghanbari Adib , Atiye Etemadi

Supervisor: Eric Knauss, Ph.D., Department of Computer Science and Engineering
Examiner: Daniel Strüber, Department of Computer Science and Engineering

Master's Thesis 2022
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2022

Challenges of Integrating Diverse Requirements Documentation and Traceability Practices: A case study in large-scale agile systems development

Kiana Ghanbari Adib, Atiye Etemadi
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

Automotive companies are moving toward in-house software development due to the increasing demand of using software in their products. As part of quality assurance for safety objectives in their software, these companies are required to ensure traceability for requirements and their related artifacts. The requirements and their corresponding trace links coming from different Agile Release Trains (ART) in the company are required to be integrated across these ARTs to ensure quality.

The differences in requirements documentation and traceability practices that are being used by different ARTs in an organization poses challenges in collaboration across these ARTs for the purpose of traceability integration.

This study is conducted in an automotive company that is applying large-scale agile in their processes. The analysis for this study is based on 15 interviews from representatives of 4 ARTs that are working together in the same department. Further, a workshop has been conducted with 18 representatives from the same ARTs to present the results along with 16 survey responses to validate the results from our interview analysis during the workshop.

This study indicates that differences in requirements documentation and traceability practices lead to inconsistencies in the data related to requirements and trace links, which impacts the process of automating traceability integration across these ARTs. Moreover, this paper identifies the differences and presents the reasons behind them. Our findings show that while there are reasons that justify the need for diversity in some parts, there are also other factors that contribute to these differences. Further, the need and suggestions for moving toward more alignment and a shared understanding of these practices across these ARTs, to achieve a common solution for the integration of traceability, is investigated.

Keywords: Computer, science, computer science, engineering, project, thesis.

Acknowledgements

We gratefully recognize the help of our academic supervisor Eric Knauss for his continued guidance, valuable inputs and suggestions. His thoughtful, detailed feedback has been very valuable to us throughout the study. We would like to thank Anastasia Koni who discovered the problem in the company and helped us through different steps in our research. We greatly appreciate her assistance during this study. We would like to extend our appreciation to Rebekka Wohlrab for accepting to be a guide for us in some points. In particular, we would like to express our gratitude to our examiner Daniel Strüber for his valuable feedback.

We would like to extend our acknowledgements to Johan Lidén Eddeland, the company's supervisor, Aboozar Zarini, the company's team manager, as well as all the participants in our interviews, survey and workshop. We also express our deepest gratitude to our family and friends for their unfailing support and continuous encouragement.

Kiana Ghanbari Adib, Atiye Etemadi, Gothenburg, June 2022

Contents

List of Figures

List of Tables

1
Introduction

Today, utilization and development of software in the automotive industry domain is evolving rapidly. This is also due to the fact that using software brings some advantages such as decreasing costs and facilitating innovation in this domain [1]. In addition, automotive companies, which are considered as large organizations, are moving toward adopting large-scale agile practices in order to upgrade their processes and remain competitive in the automotive industry market [2, 3]. In automotive companies, the systems under development are mostly safety-critical, which implies the need for compliance to safety standards [4]. In order to fulfill the safety standard for requirements, there is a critical need for facilitating tracing between related artifacts, to make sure that the required tests have been applied [4]. One big challenge that currently exists related to applying traceability, is utilizing diverse tools and artifacts, by different teams in large-scale automotive companies [2, 4].

While integration of tools is essential to facilitate traceability and is considered as one of the main reasons that motivates the demand for alignment, there are other factors like the core differences in disciplines and elicitation methods in teams working in different areas, which urge the need for diversity [5, 6]. Since, both the alignment and diversity of requirement engineering (RE) practices are required in the automotive domain, this implies the need for balancing these alignments and diversities [5].

In this case study, we focus on analyzing Requirements Documentation and Traceability Practices that are being utilized in three Agile Release Trains (ART), working in the same department in an automotive company. The Agile Release Train (ART) refers to a group of agile teams that work together to develop and present one or more solutions over time within a company [7]. The aim of this study is to apply in-depth research methods, in order to find the differences of Requirements Documentation and Traceability Practices across these ARTs and the reasons behind them. Furthermore, the impact of the differences between Requirements Documentation and Traceability Practices on facilitating requirements traceability, across ARTs, will be analyzed.

1.1 Statement of the Problem

Due to cars becoming more and more dependent on software, automotive companies need to increase their abilities to build digital systems. Software and digital artifacts behave differently from physical components, implying different life-cycles of Software. For example, it is possible to update software components late, even after the vehicle has left the factory. Thus, there is a great need to be able to man-

age requirements during their whole lifecycle in a large-scale automotive company [5]. Therefore, creating and maintaining trace links between requirements and other artifacts is important [8]. However, in the automotive industry, RE practices might differ between different organizations and even between different teams [5]. These differences make any effort to integrate the multiple tools and processes being used as well as to provide consistent traceability really expensive [5].

In order to achieve digitalisation of the automotive industry and increase its software competency, it is important to provide solutions to facilitate collaboration to deal with the gap between diverse RE practices across different areas [5]. In large-scale agile, these areas are for example different ARTs, which may develop their own set of specialized practices. Still, traceability and collaboration across ARTs must be supported to allow efficient development of a consistent product. There is a need to facilitate such traceability and collaboration and to examine and identify these differences in an industrial setting.

1.2 Statement of Purpose

Every ART has its own conventions, when it comes to Requirements Documentation and Traceability Practices. The supporting ART, which is responsible for providing frameworks and tools for the other ARTs that are developing the product, is facing difficulties in terms of understanding the differences in structures and namings related to requirement levels between these ARTs. From the perspective of the supporting ART, it's hard to provide solutions while these differences exist.

In this case study, we aim to investigate the differences that exist in the practices related to documenting requirements and traceability workarounds between these ARTs and find out why these differences exist and what is the effect of the differences on the process of traceability related to requirements.

1.3 Case Company

The case study is conducted in the case of an automotive company. Since the production and use of software in companies is developing, this company is no exception and has been following the path of using software in its products for several years to keep pace with the market. This company is also making progress in leading the production of their software toward in-house development. This has caused some parts of the organization to have a special attention to the process of software development and its life cycle.

In this study, we will investigate the case of a department consisting of three ARTs working on functions that are dependent on each other on a higher level. Moreover, there is another ART in the department, which we call the supporting ART. This ART is responsible for providing platforms and solutions for the other three ARTs. As part of the services that the supporting ART is required to provide for these three ARTs is to provide a solution to integrate traceability based on the data related to requirements and their trace links that come from each ART. We will address the problems that the supporting ART is facing to provide this solution for the ARTs.

Our main focus is to explore these three ARTs and the supporting ART to investigate their practices related to requirements and traceability and integration of the corresponding data that are affected by the differences of these practices across the ARTs. Further, we will mostly focus on these practices in relation to in-house software development.

1.4 Research Questions

In line with the purpose of this study and in context of related literature, this thesis aims to investigate how global system level traceability can be facilitated while at the same time allowing diversity in Requirements Documentation Practices on the level of local sub-areas. In the automotive case we are targeting, the global system level is represented by one department, while the local sub-areas are represented by three ARTs within this department.

RQ1. Which differences in Requirements Documentation and Traceability Practices in local sub-areas of a system are significant for collaboration on global system level?

RQ2. Which are the reasons behind these differences in Requirements Documentation and Traceability Practices between local sub-areas?

RQ3. To what extent can we support different Requirements Documentation and Traceability Practices within the local sub-areas and still trace between similar and related artifacts on the global system level?

1.5 Outline

This study is outlined as follows:

The background and related work is described in Chapter 2. In Section 2.1, we discuss the concept of Requirements Documentation and Traceability. Related work to this study is introduced in Section 2.2. In Chapter 3 we introduce the research methods that we applied for this study. Then we discuss why we chose a case study for this project in Section 3.1 and how we obtained the data in Section 3.2. The data analysis process of the study also will be explained in Chapter 3.3. In Chapter 4, we present the study results then we analyze and discuss them in detail in Chapter 5. Section 5.2 elaborates the Implications for research based on the results while Section 5.3 explains implications for practice. Limitations and delimitations of this study also will be discussed in Section 5.4. The study is concluded in Chapter 6 with a summary of the main findings and suggestions.

2
Background and Related Work

In the last few years, software has become one of the main components in car production, moving the automotive industry toward an increased focus on developing and maintaining softwares [1]. In addition, using agile approaches for software development in industry is profitable, which can increase collaboration and productivity as well as enhancing product quality [9]. Automotive industry utilizes large-scale agile methodology to an increasing extent, which implies the need for reconsidering the ways of managing their artifacts [2].

In this chapter, we will provide background to provide a better understanding of artifacts related to requirements documentation and traceability. Further, we will go through the related work to our study about diversity of these artifacts through different teams.

2.1 Background

In this section we will provide a short summary of the concepts of requirements documentation and requirements traceability and also their importance in the automotive industry.

2.1.1 Requirements Documentation

Requirements documentation is the foundation of validating different artifacts through the process of software development [32]. In order to document requirements with high quality, they must be specified in the right level of abstraction, clear and understandable [32].

Moreover, reusability of artifacts like requirements is of high value when it comes to productivity and reducing costs in the automotive industry, where time and efficiency is of great importance [25]. Need for reusability encourages the use of models or patterns to formulate requirements documentation [31].

Apart from this, it should be noted that RE plays a crucial role when it comes to the automotive industry, this requires communicating the RE practices and their evolution through the borders of the organization [31].

2.1.2 Requirements Traceability

In order to ensure safety in the automotive domain, maintaining the relations between software artifacts is required, which is defined as traceability [4]. Requirements traceability is defined as the capability of maintaining the relationships between requirements with other requirements and requirements with other software artifacts

[18, 21]. Requirements traceability provides us with the ability to follow change effects of requirements on other artifacts based on provided links [19].

Furthermore, safety standards require a complete requirements traceability, which means providing bi-directional links from requirements to the design and artifacts related to validation like test cases [20]. Moreover, requirements traceability emphasizes transparency of interdependencies between these artifacts [30].

Applying requirements traceability in an efficient way in software development, can lead to the result of improving the quality of the software and decreasing the total costs [22]. Although, the process of providing this traceability includes extra attempts related to documenting the requirements and other development tasks, while it does not provide a direct benefit and is not compelling for the stakeholders, which results in ineffective and post-development traceability [22, 23].

Due to the complexity of the system and keeping traceability of requirements, a challenging part for large-scale automotive companies could be focusing on speed and integration, which requires special management and guidance [6].

2.2 Related work

In this section, we will discuss the studies that are related to our work about diversity in different practices related to requirements and the effect of diversity on integration and traceability.

According to Pinheiro [18], automation of traceability is required due to the diverse types of data and the increase in the amount of data that is needed to provide trace links in large software systems, which can make the process of creation and maintenance of trace links more efficient. Further, he states that this data comes from various artifacts that are distributed between teams, which makes it unimaginable to be aware of all of the possible links that might exist and provide a way to view the chains of these trace links without automating the process. In our study, we will investigate the reasons that lead to challenges in automating the data extraction process for integrating traceability and how it affects the process in practice.

Kasauli et al. [23] report inconsistency in the artifacts related to requirements, coming from different teams, claiming that the level of details provided in a specific abstraction level is not compatible. The study is conducted by using interviews and other methods such as focus groups and workshops across different large-scale companies, identifying various challenges related to RE. They explain that teams have their own way and tool to specify requirements, which makes inconsistency in the requirement artifacts. This causes issues in dealing with requirements across boundaries and coordination between teams. Moreover, they add that requirements knowledge is distributed between individual teams working in parallel, with different understanding of this knowledge, this poses challenges to the management of requirements and the coordination of this knowledge through the organization. Further, they state that in order to achieve a shared understanding across the organization, there is a need to maintain documents related to requirements. They emphasize that the automotive industry has struggled with testing due to the quality of requirements, which poses the need for filling the gap related to documentation of requirements and deficiency of information, by ensuring communication between teams. The main focus of this study is finding a wide range of challenges related

to RE and as parts of their results they point out that inconsistent requirements and tools as well as a lack of a shared understanding causes lack of information that is required for testing. We will put the main focus of our study on this specific challenge of inconsistent requirements and tools and also its effect on extracting information for traceability integration.

Eliasson et al. [27] introduce finding the right level of abstraction, as one of the challenges in specifying requirements. They convey that a requirement can be considered as over-specified by one individual, while it seems under-specified to another individual, given the amount of detail that is provided in the requirement specification. They emphasize the need of finding the balance between over-specification and under-specification. Further, they claim that the current solution to achieve this balance is based on informal communication, which can turn into assumptions as a consequence of lacking time close to deadlines. Moreover, in another study, Dikert et al. [24] acknowledge that the quality assurance of the system can be adversely affected by insufficient breakdown of requirements into different levels or ambiguous requirements. Additionally, Wohlrab et al. [16] add that due to the existence of various disciplines among different teams and departments, there might be differences in the interpretation of RE concepts in large scale companies. As part of our study, we explore how the diversity of requirements documentation is related to different interpretations of the content that should be provided in each abstraction level. Moreover, we investigate how this causes ambiguities in the requirements data for integrating traceability across the ARTs.

Maro et al. [4] state that one of the main challenges in creating and maintaining trace links, is the diversity of the tools and artifacts, which results in different formats and inconsistencies in the required data. The study investigates the challenges related to traceability and provides solutions for these challenges based on a combination of different methods such as literature review and a case study in the automotive industry with a supplier company. According to the study, utilizing diverse tools causes issues with finding trace links across these tools and finding solutions for integration is of great importance. The challenge of diversity in tools will be one of the main focuses of our study. It should be noted that we will look into the tools and practices that are in direct connection with documenting requirements in an automotive company that is moving toward more in-house software development.

Wohlrab et al. [2] elaborate on alignment related to boundary objects, which are the artifacts that are common for diverse teams within a company in agile system engineering. The study focuses on system engineering artifacts, which consists of artifacts providing information and presenting knowledge related to system engineering, including architecture models as well as high-level and low-level requirements documentation and test cases. Current practices related to managing these artifacts in six large scale automotive companies are investigated. Further, a number of practical challenges in terms of managing these artifacts are defined, which includes balancing diversity and alignment, artifact degradation and difficulties in applying agile practices in teams working in non-software areas to facilitate collaboration. Finally, guidelines are provided for managing these artifacts, which emphasizes on locating boundaries as the starting point. These boundary objects are required to be communicated across teams, which can be facilitated by a tool that enables traceability links between shared artifacts managed by different teams. As mentioned by the guideline, there is also a need for flexibility when it comes to creating locally

relevant artifacts, which are the artifacts that might differ between teams.

In another research, Wohlrab et al. [5] shed light on how diversity and alignment of different RE practices is applied in different Requirements information models(RIM) in practice, by conducting a case study on four large scale automotive companies. Reasons behind the need for both diversity and alignment were investigated. The results of the study suggest the need for alignment to encourage teamwork, providing a common language for achieving better collaboration between teams, improving the quality by facilitating testability and moving toward an standard approach as the rationale behind alignment. On the other hand, in the results it is acknowledged that the need for diversity arises from differences that exist in the methods, requirements elicitation practices and the disciplines applied in various areas in the context of the automotive industry. Finally, suggestions are provided for managing RIMs in order to facilitate the balancing of diversity and alignment. In our study, the purpose is to focus on the part of RE practices that are about documenting requirements and investigate it in one automotive company in order to provide a deep understanding of the differences of requirements documentation and traceability practices in the ARTs that are working in the same department. Further, we will provide the reasons and factors that contribute to these differences, while Wohlrab focuses on the need for differences. In addition, the effect of diversity on providing traceability is just briefly pointed out in this related research and further investigation on this topic will be provided in our study.

3
Methods

3.1 Research Design: A Case Study

In order to get a deep understanding of a current problem in relation to its natural context, based on collecting data from a limited number of entities, a case study is considered as the suitable method [11, 12, 13, 36]. The adaptability of a case study makes it the right match with software related problems that are prone to complexity and fast changes [11].

The first research question (RQ 1) in our study, refers to finding the differences of Requirements Documentation and Traceability Practices between 3 different ARTs in an automotive company. In order to find the similarities and differences, we take a close look at these practices in each ART to get an overview of how they document their requirements and what are their current traceability workarounds. These practices include the tools that each ART is currently using for requirement documentations, how they structure their requirements, such as different levels that they use in order to break down a requirement and the styles of the requirements (e.g. text, diagrams). Furthermore, we compare these practices between these three ARTs to find how they are similar or different from each other.

The second research question (RQ 2) is about the reasons behind these differences. Answering this question requires a deep understanding of why in each ART, teams are using specific practices for documenting their requirements or providing trace links related to their requirements.

The third research question (RQ 3) relates to traceability and how the current differences are affecting integration of requirements traceability across these 3 ARTs. We need both the viewpoint of the ARTs and the supporting ART in this matter. We need to gather the information from RQ 2 to find the reasons that lead to diversities such as the reasons that explain the need for diversity along with the factors that are affecting the process to increase diversity across the ARTs. Then the need for alignment and customization and also suggestions that can lead to decrease the amount of diversity and address the problems that lead to diversity are discussed with the representatives from the ARTs.

3.2 Data Collection

In order to conduct a case study, developed on qualitative data, it is essential to apply triangulation in the process of collecting data by utilizing more than one type of methods and diverse sources of data [11, 37]. The data collection of this study is based on applying different methods including interviews and a workshop along with collecting survey responses for validation. It should also be noted that

before moving to the official methods for data collection, we had several meetings with representatives from different ARTs to get familiar with the tools related to requirements and getting a grasp on their conventions and knowledge about the problem.

3.2.1 Interviews

Semi-structured interviews have been employed as the basis of our data collection. Semi-structured interviews provide us with both the plan of what questions need to be addressed to get the data required for our study, and the possibility to ask the questions based on the conversation flow [11]. Additionally, the interview questions were designed with the purpose of providing data to address the research questions [11].

We started each interview by asking some general questions about the interviewee's ART, team, their role and experience related to requirements and traceability. Then we moved through different topics and asked the questions related to the corresponding topic with the aim of collecting data that can be of value for our research questions. For each topic we started with explaining the topic and related terms in order to avoid ambiguity and misunderstanding and then continued with asking open questions that could provide a wide variety of responses from the interviewees [11]. Then we moved to more closed questions that could more directly address the type of answers that were important to be addressed by the interviewees [11]. The decisions on how to ask the questions and which parts to provide more guides and possible options for the interviewees were made based on the conversation flow and the situation in each interview.

In the process of selecting the interviewees, diversity was required to be taken into account. Perspectives of individuals coming from different roles and parts of the organization is recommended to be included [11]. In order to achieve both diversity and balance in our data collection, we tried to have the same number of representatives from each ART. The final outcome was having around four representatives from each of the three ARTs and also the supporting ART. At the end of the interview process, a total number of 15 interviews were conducted.

Apart from this, we made an effort to choose specific roles and have interviewees from those roles or the most possible similar roles in each one of the three ARTs. The selected roles are tried to be diverse within the three ARTs, while keeping a similar combination of them across the three ARTs. Moreover, as far as it was applicable we selected our interviewees from different teams in each ART.

The situation was quite different in the supporting ART, since mostly 1-2 teams were focused on providing traceability for the other three ARTs and the employees who had knowledge or responsibility about the issues related to differences in Requirements Documentation and Traceability Practices, were limited to the roles of developers or architects.

The interviews were conducted in online meetings and recorded with consent of participants. Initial transcripts were generated and reviewed later after the interview by the researchers. Each of the interviews were handled in one session, with the average duration of 60 minutes. The interview guide (Appendix A) and consent form was sent to the interviewees at least one day before the interview by email. The distribution of roles, teams and year of experience about the topic in the interviews

are shown in Table 3.1.

Table 3.1: Information about the interviewees

ART	Interviewee Role	Team	Related Experience(Year)
Supporting ART	Developer	S1	1
Supporting ART	Developer	S2	2
Supporting ART	System Architect	S3	2.5
Supporting ART	Developer	S1	5
ART 1	Scrum Master	A11	7
ART 1	Function Owner	A12	10
ART 1	Tester	A13	10
ART 1	System Designer	A11	15
ART 2	System Architect	A21	16
ART 2	Product Owner	A22	8
ART 2	Product Owner	A23	6
ART 3	Product Owner	A31	3
ART 3	Function Developer	A32	9
ART 3	Function Developer	A31	1
ART 3	Scrum Master	A33	2

3.2.2 Workshop

As a next step, we arranged a workshop with the purpose of bringing different parties (representatives from ARTs) together to create an interactive group that increases the exchange of knowledge between the ARTs. By conducting the workshop we aimed to share our findings clearly with the company about different perspectives. Furthermore, we wanted the representatives from the ARTs to join together in the workshop to explore the suggested solutions given in the interviews for solving the problems through co-discussion, as well as make further reasonable comments and suggestions for related problems [26].

Moreover, this gave us the opportunity to go through the survey questions with more clarification on the concepts and decrease the risk of misunderstanding the questions by the survey participants.

We invited 27 people from the company to this workshop that can be devided into two groups of participants. The first group consisted of interview participants, while the second included the other company employees with different roles that were also from the three ARTs as well as the supporting ART who could be interested in the study's findings and provide more explanation for the questions.

The workshop took place in May 2022, in one hour session, in combination with an online survey, with 20 attendees including 2 researchers and 18 company representatives. Because some employees work from home and prefer to participate in an online session rather than in person, we decided to do the workshop online through Microsoft Teams in order to increase the number of participants. In addition to conducting the online workshop, it assisted us in implementing an online survey step by step during the session so that participants could quickly respond to the survey from their desk.

We started the workshop with an overview of the study and its purpose. After that we presented our findings in three sections throughout the workshop. The first section dealt with RQ1 and we concentrated on presenting the tools for requirements documentation, requirements abstraction levels and traceability workarounds and their comparison across the ARTs.

In the second section, we presented our results concerning the reasons for the differences and current problems based on differences which are related to RQ2. Finally, section three dealt with RQ3, in which we discussed the necessity for ARTs practices to be aligned, as well as suggestions that we got from interviews.

At the end of each section, participants were asked to reply to particular survey questions. Then we requested attendees to complete and submit the survey at the end of our presentation, after that we discussed the results together to provide some interaction and knowledge sharing.

To keep track of the limited time of the workshop, we discussed some of the questions that seemed most interesting to the participants, as well as the questions that most participants answered with a strong agree or disagree option.

In the next section, we'll go over the survey in further detail.

3.2.3 Survey

To obtain data from other participants within the company, as well as validate the interview results, we decided to conduct an online survey as the final method [13]. This gave us the opportunity to check our findings that were based on our interpretation from the interviews with our interview participants and other employees. We used Microsoft Form to design the survey and we presented the survey with 14 questions during the workshop for the company. We asked our workshop participants to respond to the relevant questions step by step on various topics after getting our explanations on that topic during the workshop. 18 of the 27 participants that were invited to the workshop session showed up in the session and 16 of them completed the online survey.

The survey questions (Appendix B) were conducted based on the coded data and findings provided from the previous steps, in order to validate our results. Some of the questions were designed based on the analyzed codes from the interviews about differences in requirements documentation and traceability practices across the ARTs. So, we could get more information from the audience and also ask them to help us to identify if there was misinterpretation from our side based on the interviews. To help us better quantify the level of agreement among the participants about the topics, a large percentage of the survey items used a Likert scale ranging from Strongly Agree to Strongly Disagree.

On the survey, we inquired about the following topics, and it was critical for us to determine the participants' level of agreement on these issues:

- Reasons behind differences in requirements documentation and traceability practices
- The need of moving toward more alignment in requirements documentation and traceability practices across the ARTs
- Suggested solution for the tools problems
- The need of a reference/ standard model for requirements and traceability

We generally received agreements on the suggestions that were based on our interview analysis, from the workshop participants on the survey (Appendix C). Therefore we can state that the survey results confirm the interviews results to a high degree.

We sent the result of the survey to the participants from the company the day after the workshop. Further, we have attached the result of the surveys to our report (Appendix C) for future access [14].

3.3 Data Analysis

As it was mentioned in Section 3.2.1, we conducted 15 interviews in order to collect data for our study. In order to prepare the interview data for analysis, first we needed to have a reliable transcription of the interviews.

We can see transcription as a way to get a better understanding of what has been said during an interview, which can be of value during the process of data analysis [33]. The transcriptions of the interviews were reviewed by both the researchers and checked with the records in order to fix the mistakes. This was also considered as an opportunity to review the interviews and comprehend it better before moving to the next steps.

The names of the interviewees were removed from the transcript files and replaced by the corresponding code that was assigned for each interviewee, so that the transcripts would be kept anonymous.

In order to get a better understanding of what we have in the transcripts, which consists of huge amounts of data and transform it to information with transparency that can be presented in a report, coding was applied based on the transcripts data [34, 35].

Data analysis was followed based on predefined steps [15].
Before starting the coding process, we exported the transcripts text for each interview into an excel file. One important factor in terms of qualitative data analysis is to keep the chain of evidence, which means that everything should be traceable [13]. To achieve this traceability in our data analysis, we kept the data and all the assigned codes in the same excel file. This way, we were able to trace which code is coming from which part of the data.

In the first step of coding, some parts of the transcripts were mapped to a corresponding code (Appendix D), which are just a shorter description of that part that gives meaning to it based on the study purpose [33, 34]. So, the amount of data that we needed to apply our analysis on, was shrunk to codes that are of value for the study [33, 35]. We went through the data in the excel file containing the whole transcript and assigned codes to any piece of data that was of interest for our study. The codes were just a summary that could help us to understand which sort of information each part of the data is referring to.

The second step of coding was providing themes for putting a group of codes that are similar in some way together [34].

Themes are used based on our inferences from the codes, coming from the first step, so that they bring the codes to a higher level of abstraction that convey concepts that are related to the study purpose [33, 34]. We provided a new column to assign categories to our codes and put the similar codes in terms of the value that they could provide, related to our research questions, in the same category.

Due to the inconsistency of codes in the emergent coding approach, the codes were needed to be modified to reach some sort of compatibility [34]. We made a copy of the results of our coding from this step in another excel file, to find the similar codes for the coded interviews. The codes that conveyed the same meaning were merged to achieve more consistency and make it easier to count and find the codes that were more repeated (Appendix E).

Finally, implications of the themes from the previous step were provided that could be directly connected to a particular concept of the study[33]. For the next step, we brought all the codes from all the interviews together in an excel file. All the codes were reviewed and merged based on their concepts. One of the purposes of coding is mapping the data in order to find the relations with research questions, so we can find the parts of the data that are related to a specific research question [34]. We mapped the codes to different subsections related to our research questions and counted them to find which codes are more repeated in our data (Appendix F).

The process of coding mainly consists of a few iterations to revise the codes and validate them based on the knowledge we gain through the process and make them more consistent [34]. These iterations and revisions were applied in the whole steps of our coding process.

Following the completion of the coding process, we created a Miro board to show the codes associated with each study topic and also find the better connection between them and a better structure to present our results.Miro is an online collaborative whiteboard with a variety of features and themes that greatly aided us in effectively visualizing and managing our coding results during the final steps of our study.

We placed sticky notes to display codes and recategorize them in an attempt to merge relevant codes as much as feasible to a more abstract topic. We used various blocks and assigned each one a high level label, then searched for more related codes and filled the block with them. We also created 4 frames which consist of several blocks. The frames were chosen in response to the research questions' main topic (Appendix G). The frames are as follows:

- *ARTs Similarities and Differences:* This frame consists of four main blocks for the three ART as well as the supporting ART. Moreover, each of the blocks has several sub blocks with relevant codes. For example, block ART 1 consists of 7 sub blocks which are the ART's requirement documentation tools, abstraction level and style of requirements that are stored in tools, types of trace links, traceability workarounds, etc. We can easily notice the similarities and differences of the ARTs in this frame by comparing the similar sub-blocks of each ART.
- *Similarity and Differences Reasons:* In this frame, we collected different blocks for reasons behind the ARTs similarity and differences
- *Problems:* We have two blocks in this frame for problems that are based on differences and other problems related to traceability. We attempted to connect problems from these blocks to related blocks in the Similarity and Differences Reasons.
- *Suggestions:* In this frame, we organized all of our interviewees' suggestions related to solving problems, supporting reasons for alignment and its challenges and suggestions in different blocks.

While using the Miro board, we put the number of occurrences of each code in each sticky note (which are the final codes from the excel files) as a factor to help us identify which problems are mentioned from more interviewees. However, we tried to avoid ranking codes based on only this factor and tried to include all the codes in our interpretation and further validate them by the survey results.

After finishing our journey of data analysis, we designed a survey and a workshop for validating the analyzed results, in the company.

As previously stated, we held a workshop and the participants were asked to complete a survey during the session. The survey results helped us to validate our interpretations and review the high level codes and bring them more confidently to our results.

4

Results

4.1 Challenges for Automating Traceability Integration Across the ARTs

In order to integrate traceability across the ARTs, the supporting ART needs to extract the data of the requirements and trace links related to requirements from each ART. In this chapter we go through the factors that impact the process of extracting this information. First, we show our results related to how the differences in requirements documentation and traceability practices between these three ARTs are affecting the automation of this process. Then, we go through other problems in accessing the required data based on missing data related to requirements and trace links. Figure 4.1 provides a summary of the main challenge and the factors that contribute to this challenge.

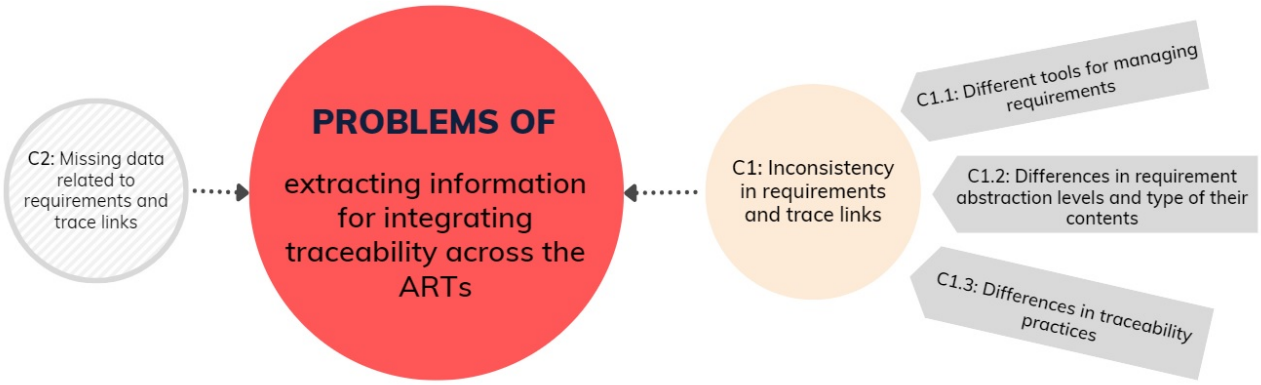


Figure 4.1: Summary of challenges for automating the integration of traceability across the ARTs

It should be noted that the main focus of our study is on the problems that arise from the differences of requirements documentation and traceability practices across the ARTs.

4.1.1 C1: Inconsistencies in Requirements and Trace Links

Requirements and trace links related to requirements are inconsistent. This is partly due to the fact that there are differences in the ways the ARTs are managing their requirements and trace links related to requirements. These differences are mostly

coming from using different tools for documenting information related to requirements and their trace links. These trace links consist of links that connect requirements to other requirements or requirements to other artifacts (e.g. test cases, implementation). In addition, the structure of requirements such as abstraction level and the type of content in these levels differs between the ARTs. Moreover, the requirements traceability workarounds and the type of existing trace links even differ within one ART.

These differences induce inconsistency in data related to requirements and trace links. Furthermore, they bring the data to a state that can not be understandable outside the ARTs. Also, the diversity of practices leads to a situation where the supporting ART has no choice but developing and maintaining different solutions for each ART in order to extract the data related to requirements and their corresponding trace links, if they want to provide this data extraction automatically for the ARTs.

Consequently, the current workaround to avoid maintaining different solutions for each ART is a CSV template file, provided by the supporting ART, that needs to be filled out by each ART. The manual option of using these CSV files requires too much effort by each ART to prepare the required data, which is not efficient.

The following sections explain the identified differences that are giving rise to the problem of inconsistency in requirements and trace links.

4.1.1.1 C1.1: Different Tools for Managing Requirements

All the ARTs are required to use a new official tool for documenting requirements, which has been introduced in the past few years. However, our findings show that the start date of the transition has not been the same for different teams, even within the ARTs. Moreover, in some teams the old official requirement documentation tool is still in use for the projects which are in the maintenance phase. Furthermore, some teams only managed to make the transition to the new tool for the new projects recently.

Apart from this, other tools are being used to store information related to requirements.

In all the ARTs, employees stated that they are using Simulink, whether to store unit level requirements or connect implementation from Simulink to requirements in other tools, such as the requirement documentation tool.

It should be mentioned that addressing the dependencies between requirements that are spreaded between these tools is an ongoing issue for the three ARTs and the supporting ART.

Furthermore, in ART 1, a customized version of Jira for the company was mentioned as a tool they use to reference requirements for keeping track of the status of requirements and assigning tickets for reviewing them.

Also, in ART 2, Medini is being used as a tool for safety analysis and metrics evaluation related to hardware, which requires it to be linked to software requirements in the official requirement tool.

4.1.1.2 C1.2: Differences in Requirements Abstraction Levels and Type of their Contents

The abstraction levels of requirements differ between these three ARTs. These differences are in the levels that each ART is using for the breakdown of their requirements and the type of content in the same levels.

Based on our findings, in ART 1, requirements are broken down into more levels compared to ART 2 and ART 3. Apart from this, due to the direct user interaction with the function provided by ART 1, they have a requirement level related to the user expectation from the product. This level is supported by use case diagrams for more clarification of the customer needs. Further, they break down these requirements into other levels, moving from higher levels of abstraction to lower levels and finally they end up breaking down on software requirements. All these levels are stored in the official requirements documentation tool.

Although, in ART 2 and ART 3, the final user does not interact with the functions directly and they skip the requirement levels related to the final customer in the way of breaking down their requirements. In ART 2 and ART 3, system level and software level requirements are currently being used. These two levels are stored in the official requirements documentation tool. In addition, both of these ARTs have moved toward writing more higher level contents on the system level.

Apart from this, in ART 2, the software requirements are broken down further into unit level requirements. These requirements are on the implementation level and stored in Simulink. In addition, in ART 3, the team working with diagnostics only uses software level requirements.

While all the three ARTs are using software level requirements in the official requirement tool, the types of content provided across the ARTs are not consistent. For instance, ART 1 is focused more closely towards the implementation level compared to ART 2 and ART 3. In fact, in both ART 2 and ART 3, the participants reported trying to avoid writing implementation level requirements on this level in the official requirement tool.

Furthermore, all the three ARTs are using text supported by diagrams which are added in image format to provide content for their requirements. The reason for adding diagrams(e.g use case diagram, sequence diagram) is to provide better understanding for the textual requirements. The following quote provides an example of the requirements style:

> *"The level of requirement can be the same but it can be written in different ways. Some use more diagram styles and state flows, flow charts, etc. I'm writing text perhaps. Others have more use cases based on the customer function and not so much text, so it's a bit of a difference."*
>
> — A function owner

The above quote from a function owner, working in one of these three ARTs, illustrates that the combination of text and quote can be one of the sources of inconsistencies in the content of requirements.

4.1.1.3 C1.3: Differences in Traceability Practices

There is no unified way of providing trace links related to requirements even within the ARTs. We have identified diverse ways of providing trace links in different teams. In fact, there is a lack of having a stable official way of providing requirements traceability in the ARTs. Apart from this, all the identified current workarounds to handle requirements traceability are based on manual solutions.

For example, adding the requirement id or a hyperlink from the official requirements documentation tools, in Simulink has been one way to provide trace link from these requirements to implementation in Simulink. Some of our participants from ART 3 said that they have used this approach at some point but they are not using it any more, while it has been brought up as an in-progress workaround from one of the participants in ART 2.

A further instance of this is that to create trace links between requirements that are in the same level or from different levels, in the new official requirements documentation tool, reference numbers have been used. These references are added as part of the requirements text to show if a dependency exists in the selected requirement with other requirements. A problem that exists in using this approach is that finding links in both directions is not straightforward, because when a requirement is added as a reference to another requirement's text, the link is not visible in the referenced requirement itself.

Moreover, there have been other workarounds to manage traceability in the new official requirements documentation tool, but due to our participants, these workarounds are not stable and are prone to frequent changes.

This should be noted that the trace links related to requirements are scattered in different tools and documents. Therefore, this leads to inconsistency in the data related to trace links, which makes finding the related links a challenge that further makes problems on the integration level.

4.1.2 C2: Missing Data related to Requirements and Trace Links

Requirements status: The features in the official tool for documenting the requirements are not sufficient for managing the status of the requirements (e.g. defined, implemented, tested, reviewed). For example, as mentioned before in one of the ARTs, the workaround is to use a customized version of Jira tickets to follow the state of the requirements. This is due to the fact that the official requirement tool can not fulfill their need to manage the status of the requirements. Accessing the data related to requirements status is problematic and challenging due to the fact that there is no integrated tool to manage their status.

Version control for trace links: The trace links are not version controlled. In fact, the history of changes in the trace links are not visible, provided that there is no way to track the changes of these links. For example, if a link is removed by mistake there is no chance to discover that a mistake has happened.

Missing trace links: We have identified traceability as an ongoing work in all the three ARTs. Incomplete or missing trace links have been reported in these

ARTs in different sections like trace links between requirements to requirements in other levels or in the same level, requirements to other artifacts (e.g. implementation and test cases).

4.2 Reasons behind Differences in Requirements Documentation and Traceability Practices

As the differences in requirements documentation and traceability practices is identified as one of the main challenges in gathering information for requirements traceability integration, in this section we will go through different reasons that are giving rise to this diversity. The reasons are categorized to tools, guidelines, problems in applying large scale agile, priority of traceability and different needs.
A summary of these reasons and their contribution in causing differences is provided in Figure 4.2.

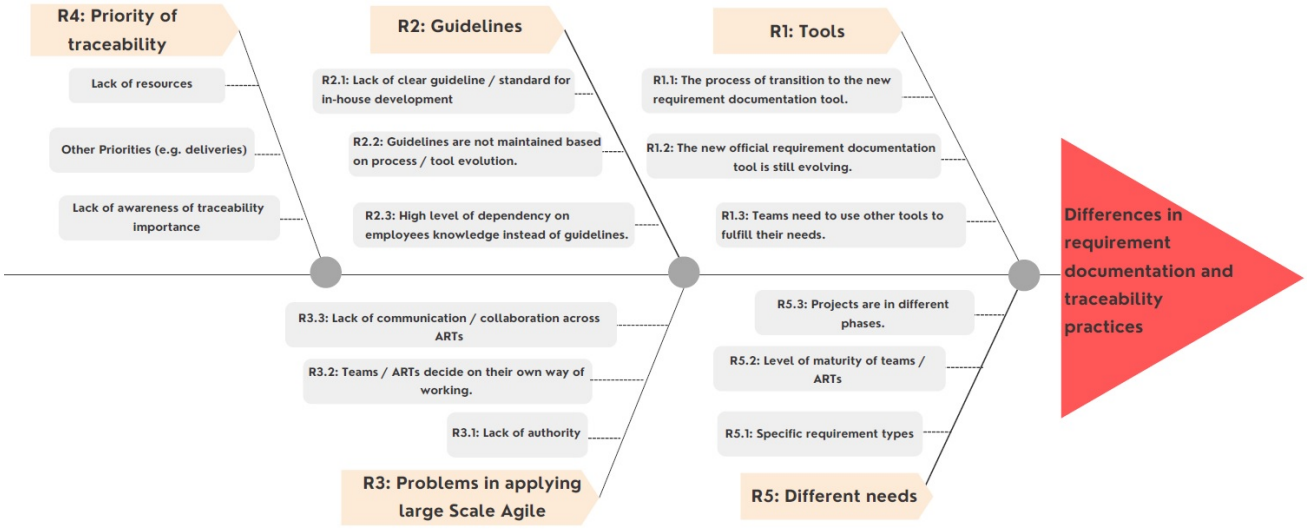


Figure 4.2: Summary of the reasons behind differences in requirements documentation and traceability practices

In addition, the results of the survey shows the level of agreement from the participants of the workshop with the reasons that end up to the diversity of these practices (Figure 4.3). This should be mentioned that the participants were asked to answer this survey question during our presentation right after our explanation about this specific topic, to mitigate the risk of misinterpretation.

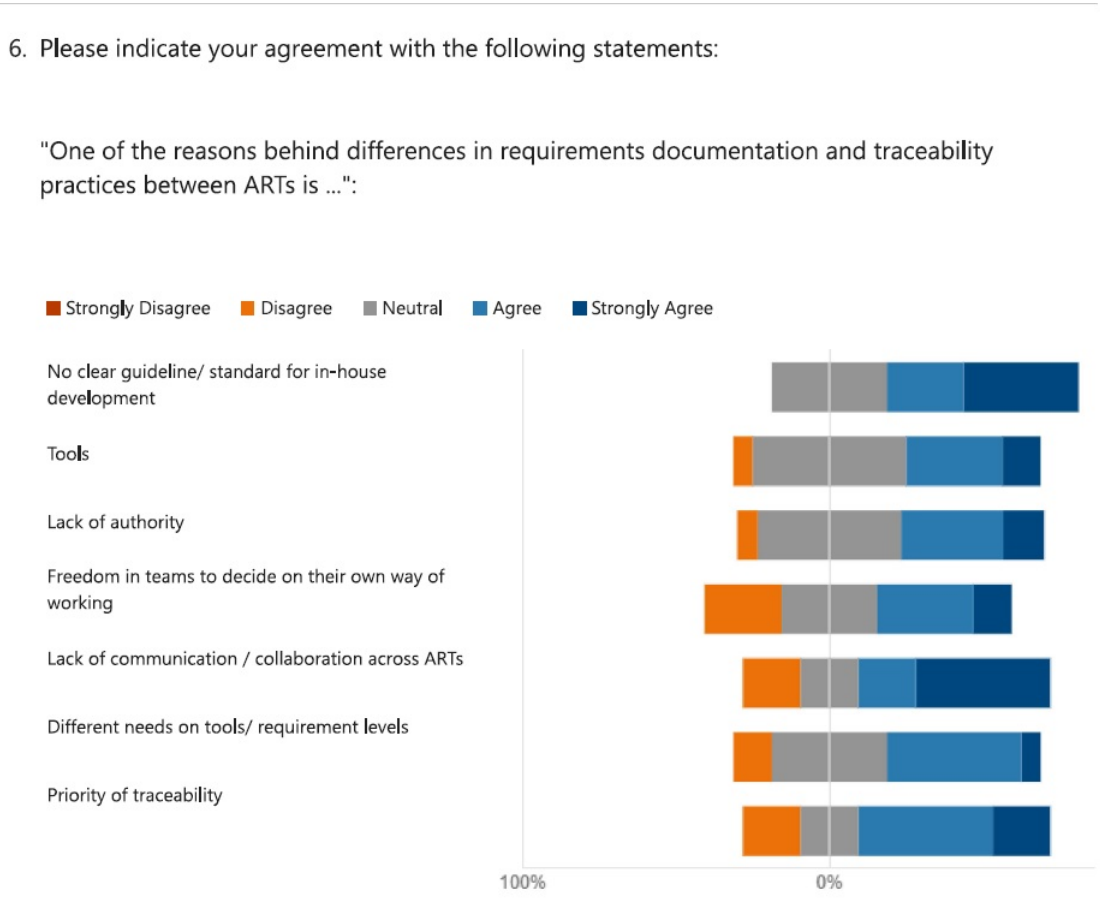


Figure 4.3: Survey results for reasons behind differences

4.2.1 R1: Tools

Our findings suggest that tools have been one of the sources of diversity across the ARTs. This related to the tool itself and the process of changing tools throughout the company. In this section we will discuss the causes of diversity that are somehow related to tools.

4.2.1.1 R1.1: The Process of Transition to the New Requirements Documentation Tool

Having the same tool across the company can bring the expectation of moving toward more alignment. However, the transition phase from the old tool to the new one has introduced new inconsistencies in the requirements. Representatives from different teams mentioned different dates of starting this transition which has caused more diversity in the tools that are employed by different teams in a specific period of time. In addition, some old projects in the maintenance phase are still using the

old tool and some teams are still relying on the old tool for fulfilling their needs to some extent. In other words, this transition process has been identified as one of the significant sources of diversity across different teams.

4.2.1.2 R1.2: The New Official Requirements Documentation Tool is still Evolving

Our participants claimed that they get frequent requests to change their way of working with the tool, especially with the functionalities related to traceability. This is due to the fact that the new tool is still evolving. Indeed, the process of updating the current workarounds and adapting to new changes is time consuming for the employees. It should be mentioned that even in one ART, the process of this change can vary between different teams. The time that a team needs to get the changes and adapt to them is not the same as the other teams. Accordingly, this can cause more diversity in the ways of working even within one ART. The traceability workarounds are not still stable due to these changes and this brings disappointment and reluctance to apply the received workaround due to the fact that the requested change would not provide long term value.

The following quote provides an example of the amount of changes that the employees are facing related to the requirements documentation and traceability practices:

> *"[...] So you're seeing the changes that come to you are too fast that you don't have enough time to adapt to it. And you think if you change it, maybe tomorrow there will be another change. You again have to modify it. "*
>
> — A function owner

As it was noted in the above quote, the high frequency of updates is considered as an issue by the employees.

4.2.1.3 R1.3: The New Official Requirements Documentation Tool is not complete

The requirement documentation tool is still missing essential features and can not provide the ARTs with all the functionalities they need for managing their requirements. In order to meet their needs, teams are utilizing other tools along with the official requirement tool. For example, we have discovered dependencies on the old requirements documentation tool in some parts. Further, the participants indicated that this tool is not sufficient in providing the required traceability, causing using other approaches in applying trace links.

A further instance of this is that in ART 1, a customized version of Jira tickets has been used for managing the status of the requirements (e.g. implemented, tested, reviewed) due to the lack of this functionality in the new official requirement tool.

4.2.2 R2: Guidelines

Guidelines were mentioned as one of the most important factors that affect diversity by our participants during the interviews and workshop. This is due to problems

in creating and maintaining guidelines that can be used as a reference point to help with clarifying different requirements documentation and traceability practices. Problems in managing guidelines can lead to misinterpretations and lack of a shared understanding that affects communication and collaboration across the ARTs.

4.2.2.1 R2.1: Lack of Clear Guidelines/ Standard for In-house Software Development

Most of our interviewees stated that they are not aware of any specific guideline or standard across the ARTs that clarifies how they should break down the requirements, what type of contents are required on each level and how to manage trace links, when it comes to in-house development. Some of our interviewees described that there are some guidelines that are provided within the ART or team that they are working in.

The following quote is related to requirements documentation and traceability practices for in-house software development:

> *" [...] So the levels of requirements and how requirements are returned and how they're linked and so on are not standardized."*
>
> — A product owner

The above quote indicates that there is no standardized approach in documenting requirements and trace links across the ARTs.

The following quote provides another instance that more specifically suggests the lack of guidelines as an important issue that leads to diversity:

> *"I think the main issue for us is that we don't really have a process on how to work with the safety software in house. And that's why we had to start with doing this requirement structure ourselves, because normally it should be in a process of how to refine requirements."*
>
> — A system architect

As it was noted by this system architect from one of the ARTs, not having a standard or instruction related to in-house software development leaves the ARTs with no choice but to define their own way in structuring requirements.

Further, only two of our interviewees mentioned that they are aware of a company level guideline that addresses the generic level of requirements. However, they also confirmed that this guideline doesn't provide clear information about the purpose of the content in a specific level and is not sufficient to clarify how they are required to provide requirements and trace links.

Lack of awareness of how to work has been confirmed by 13 of our interviewees and got the highest level of agreement in our survey as a reason for the current differences in requirements documentation and traceability practices across the ARTs. In fact, the participants complained about not having a guideline to help them to better understand how they are supposed to manage tasks related to requirements.

Apart from this, there is a lack of assessment based on standards/ common expectations on the provided documents on requirements and their trace links. Getting effective assessment and feedback is essential to help the employees in improving

their knowledge about the way of working and the outputs they produce for requirements that will be used for traceability integration later on through the process.

4.2.2.2 R2.2: Guidelines are Not Maintained based on Process/ Tool Evolution

While there are frequent changes in the new official requirements documentation tool, there is no updated guideline available for the ARTs based on these changes. Along with the tools, the process is also evolving. This evolution is more obvious in the company's plan to move toward more in-house development, which requires the current guidelines/ standards that are mostly designed for communicating requirements to suppliers to be extended for in-house development .

Problems in managing the guidelines/ standards to be compatible with the process can lead to a direction where the employees will need to deviate from the existing guidelines to be able to make progress in their tasks. The following quote is an example of this problem:

> *"[...] But maybe more that the process has kind of evolved. So, then if I see that there is something wrong in that guideline. According to how we think we should work today, then I go in and customize [...]."*
>
> — A function developer

As the above quote indicates, for guidelines to be effective, they are required to be maintained based on the tool and process evolution. Otherwise, as time passes the employees would need to deviate more and more from the guidelines and the guidelines will lose their efficacy. Most importantly, the deviations would happen in different teams and ARTs and these groups would go toward different directions in their requirements documentation and traceability practices, which contributes to more diversity in these practices.

4.2.2.3 R2.3: High Level of Dependency on Employees Knowledge instead of Guidelines

There is too much dependency on the individual's knowledge of the ways of working with requirements and traceability. Since there are no clear guidelines helping employees with how to manage requirements and traceability for in-house development, individuals in a team would be dependent on the other team members' knowledge about the related practices to a high degree. Accordingly, when someone leaves, the team will face challenges in maintaining this knowledge. The following quote is an instance of this knowledge loss:

> *"There are key people being used as a contact point for something because of their role. But, six months down the line these people might not even be here, or they might switch the role. [...] So they're not the authoritative figure anymore. Even if they have some knowledge and they can tell you, oh, this is how I used to do it back six months ago. But this is also not correct because what about now"*
>
> — A developer

As this developer from one of the ARTs explains, relying on individuals to get the required information ends up to a point when the information is not accessible or trustable any more.

Further, losing knowledge about the requirements documentation and traceability practices along with lack of a clear guideline to follow, will leave people with no choice other than relying on the existing information and their own interpretations. Consequently, different interpretations lead to more diversity in requirements documentation and traceability practices.

4.2.3 R3: Problems in Applying Large-Scale Agile

Applying agile in large-scale can bring challenges and lead to differences. Lack of authority, freedom in teams to choose their practices and also problems in collaboration across the ARTs has been identified as reasons that are promoting diversity.

4.2.3.1 R3.1: Lack of Authority

According to our interviewees, there is no role who you can refer to for providing directions on how requirements and traceability related tasks are supposed to be managed and what types of output is required to be presented by teams across the ARTs. In addition, there is currently a lack of an authority or a defined role that has the responsibility of providing feedback on the artifacts related to requirements and traceability.

In the following quote, a tester from one of the ARTs emphasizes on the changes that have happened due to the transformation to large-scale agile:

> *"[...]You changed the roles and the definition of those roles, and there are things that are falling in between. Things that need to be done that don't have an owner, [...] We don't really know who should follow up if we have these things in the report that we need to have."*
>
> — A Tester

As it is stated in the above quote there is no one responsible for the output related to requirements tracing.

Lack of an authority as a reference point to provide guidance for the employees will push them to fulfill their tasks based on their own interpretations of different concepts and practices. As a result the provided output will be of diverse quality based on different understandings.

4.2.3.2 R3.2: Teams/ARTs are Free to Choose their own Practices

Currently, ARTs are free to define their way of structuring requirements. Although, there are some conventions leading them to use almost the same requirement abstraction levels within the ARTs.

It should be mentioned that all the teams in the company are obliged to use the new official requirement tool as their main tool. But, they also have the choice to add other tools to their processes or store low level requirements in the implementation tool or the official requirement documentation tool.

The following quote emphasizes the high level of freedom in ARTs to choose their practices:

> *"Different ARTs are completely free to have their own approach, and they're completely free to change their approach at any point in time."*
>
> — A product owner

As the above quote indicates the practices can freely be chosen or replaced by new practices. While this freedom can bring agility and comfort to change the way of working to a more suitable way, it can also be one of the sources of diversity across the ARTs.

4.2.3.3 R3.3: Lack of Communication and Collaboration Across the ARTs

Most of our interviewees mentioned that they are not aware how the other ARTs are dealing with requirements and traceability. Although, based on our analysis, there are some discussions between the ARTs about their requirements and traceability practices. In addition, the way some ARTs have been managing these practices have affected the other ARTs to some degree. But, we did not discover an organized way of facilitating communication and collaboration between these ARTs.

The following quote emphasizes the lack of collaboration across the ARTs:

> *"I think generally that we collaborate too little. From my perspective, everybody is going their own way, and we are more or less discussing the same questions, but in parallel teams, it's not so efficient. "*
>
> — A system architect

As the above quote indicates, lack of collaboration can lead to losing the opportunity for knowledge sharing which can help with solving problems in a more efficient way.

The following quote directly states that lack of communication is one of the main reasons for diversity in practices:

> *"I think the reason why there's differences is that everyone works in their own bubble. In my opinion, it's just a lack of communication between teams, sometimes between teams within the same Art, but mostly between Arts."*
>
> — A developer from the supporting ART

As it was stated by this developer, there is a lack of communication between different units in the organization that is causing differences in their practice. In fact, there is a lack of awareness in the ARTs about the practices of other ARTs, what problems they are facing related to their practices and if they have specific solutions for a common problem.

Moreover, an effective way of communication can lead the ARTs to find the best practices and it might be a start point to start collaborating on promoting the best practices across the ARTs, which can lead to more alignment.

4.2.4 R4: Priority of Traceability

According to our interviewees, other priorities such as deliveries in a situation where there is lack of resources, causes the traceability related tasks to be delayed. Lack of resources has been mentioned in several interviews as one of the reasons for the existing differences and problems related to requirements documentation and traceability practices.

Further, there is a lack of awareness about the high importance of traceability. The following quote is an example of this lack of awareness:

> *"We don't really have understanding from the designing departments that are writing the requirements, how important these links are."*
>
> — A tester

As it is stated in the above quote, the teams that are responsible for managing the requirements are not informed about the importance of providing trace link for these requirements. This could affect the quality of requirements and trace links that are considered as inputs for traceability integration across the ARTs.

Increasing the compatibility of the data across the ARTs requires attention and resources to be provided from a higher level. While lack of attention to this matter will lead to more incompatibility through time due to the increasing amount of data.

4.2.5 R5: Different Needs

Having different needs is one of the reasons that justifies diversity across the ARTs. There are differences that are inevitable and required to exist, otherwise the specific needs of the teams/ ARTs can not be addressed. These differences exist in the type of requirements that the teams/ ARTs are working with, the maturity level of a team/ ART and differences in current phases of projects. In this section we will explain how these conceptual differences are leading to diversity in requirements structure and tools.

4.2.5.1 R5.1: Specific requirement types

The types of requirements that are being used can affect the abstraction levels that are needed to be able to break down these requirements to more applicable levels.

For example, in areas working with safety requirements, lower level of requirements might be essential. In the case of ART 2, we can see that they are using lower level requirements compared to ART 3 to fulfill safety requirements.

The following quote gives us a further instance for having specific types of requirements within one ART:

> *"We don't own any system, diagnostics is more like a part of every system. So it's more like, we could connect some of our software requirements towards other teams' system requirements. So probably more like that than having its own system requirements."*
>
> — A scrum master

As it was mentioned in the above quote, the team working on diagnostics does not need to have requirements on system level due to the nature of their requirements.

Furthermore, ART 1 is the ART with functionalities that are more directly connected to the final user. To address this, they need to have requirements on a higher level compared to the other ARTs in order to apply the user expectations and show what is required from the final users' perspective.

The following quote addresses the need for customization in writing requirements:

> *"If there are certain differences in what they contain, my understanding would be that the main reason is that the teams have different sorts of functionalities that they probably work with. [...] So the way we word our requirements, the way we use images or flowcharts could be different from the way they do it. So those small kinds of customizations I think definitely should be allowed [...]"*
>
> — A function developer

As it was stated by this function developer, the functionalities that the teams are working with might be different which obliges the need for different ways of writing requirements. Indeed, these differences in functionalities are more significant across the ARTs which will lead to higher levels of diversity.

Apart from this, some of the differences in tools that are being used by the ARTs can also arise from their different needs. For example, in ART 2, working with more safety critical requirements results in using a tool like Medini as an extension to the official requirement tool for safety analysis.

4.2.5.2 R5.2: Level of Maturity of Teams/ ARTs

Level of maturity of teams or ARTs also can motivate their need to consider having lower level or implementation level requirements. Maturity level can be related to the time that the team members have spent in the same domain. In other words, it depends on their knowledge related to the characteristics of the domain they are working in.

The following quote is an example of the effect of the maturity level of a team in their need for having low level requirements:

> *" [...] If the coder is completely new and has no knowledge of the standards then they need requirements. They don't really know what to code otherwise. They don't know how the system behaves and how different standards work. Then obviously the only way to get that is through requirements."*
>
> — A system designer

As it was mentioned in the above quote, while in more mature teams people might not need to have implementation level requirements and they prefer to have freedom on the lower level, for teams that are less mature they might need to have this type of requirements to be assisted on what exactly they need to provide.

4.2.5.3 R5.3: Projects in Different Phases

A team can be part of a project that is in a different phase compared to the project that the other team is working on. Applying changes in the setup of the projects such as the tools that are being used for development or requirements becomes more challenging as time passes. For instance, applying changes in tools or upgrading tool versions becomes more challenging when most part of the development is already done. In general, projects that are in the maintenance phase would have the lowest priority compared to the other projects and they might not be even included in the process of updating the tools.

The following quote gives us an example of the challenges in updating tool versions:

> *"If they use different versions of a tool, let's say Simulink. It's hard for us to force everyone to use the same tool, even though we're trying. But it might be that what one team might work on is on a product that has been in development for like 2 years. And it's really hard to just tell them you need to update your version of Simulink or MATLAB because then they might need to redo a lot of things they've already been doing. They don't want to do that, so they are more or less stuck in this old version until they go on working on a new project. "*
>
> — A developer from the supporting ART

As it was stated in the quote, requesting teams to change their tool versions through the development phase can be quite challenging. This is one of the reasons why different versions of Simulink, which is currently being used for linking implementation to requirements in other tools or storing low level requirements, exist across the teams. Another instance of this can be moving from the old official requirement tool to the new one that has been affected by the current phase of the projects.

4.3 Moving toward more Alignment in Requirements Documentation and Traceability Practices

The majority of our study participants have a positive opinion about moving to a more common/ standardized way of working related to documenting requirements and traceability across the ARTs. A more common way of working can bring advantages for the three ARTs and the supporting ART in terms of having a better understanding between them that can facilitate providing common solutions on managing requirements and traceability integration by the supporting ART.

4.3.1 Supporting Reasons for Moving toward more Alignment

During our study, we discovered that there are several supporting reasons for the need of moving toward more alignment across the ARTs.

The survey results (Figure 4.4) show the level of agreement from the participants of the workshop with the need for alignment and the reasons supporting this need.

Further, in this section, we will go over these supporting reasons for alignment.

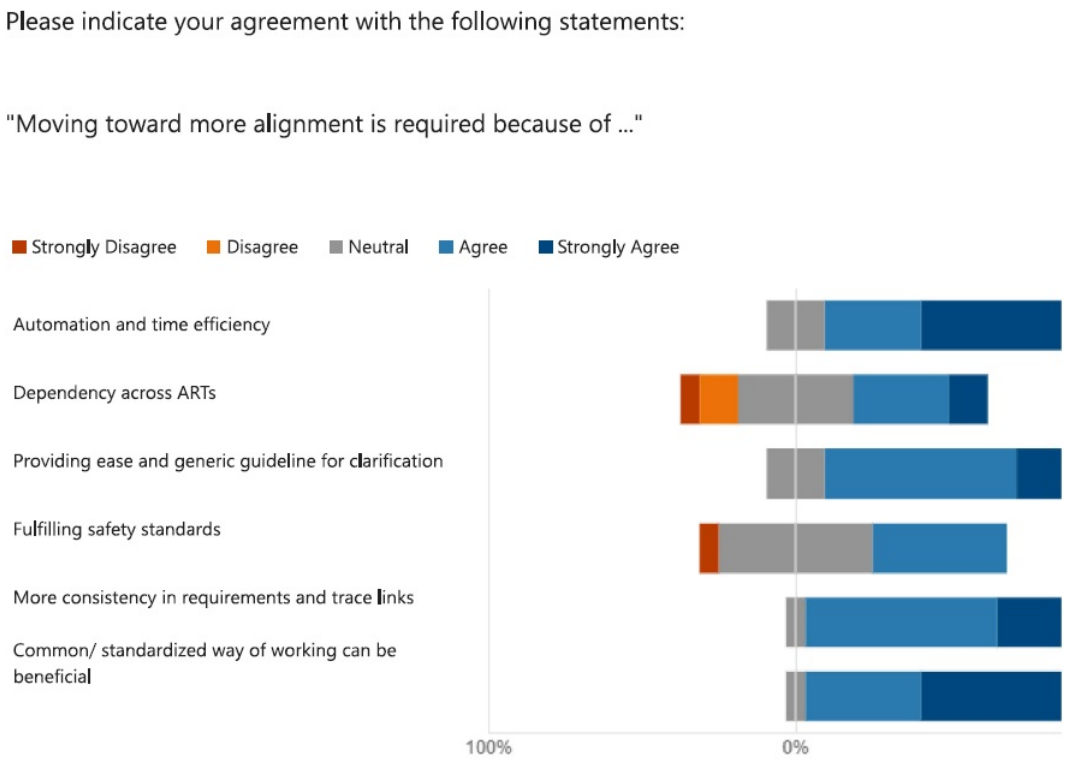


Figure 4.4: Survey results for moving toward more alignment

4.3.1.1 More Consistency in Requirements and Trace Links for Integration

As it was mentioned before, the current workaround for extracting data from these three ARTs is to ask them to fill out a CSV file manually. The problem of not being able to extract information required for traceability integration in an automatic way arises from using different tools, having different abstraction levels and also different types of contents in each level across the ARTs. The following quote confirms that diversity can affect integration:

> *"I think we will have a lot of obstacles there, especially for example in the core system where we have so many different teams and different functions and if they do this in different ways I think integration will be very difficult."*
>
> — A system architect

As it was indicated by this system architect, due to the number of teams in the ARTs, if every team wants to have their own practices this would adversely affect integration across the ARTs.

The following quote from a developer in the supporting ART emphasizes on the need for standardization:

> *"The problem here is that other aspects of their work need to be standardized. And a lot of output from their work needs to be standardized and in order for us to help them produce or automate something further. They have two choices: they even either have to comply with the new standard way or that everybody is adopting or they won't get what they need. "*
>
> — A developer from the supporting ART

As it was noted in the above quote, standardizing the outputs of requirements and trace links from the ARTs is necessary for achieving automation in the process of integration. Otherwise, the ARTs would not be provided with the solution from the supporting ART for integration.

Therefore, a standard way of working across the ARTs can help automating this process of extracting the required data and reducing manual and time-consuming work, by providing more consistent and understandable data from requirements and trace links for the supporting ART.

4.3.1.2 Dependency Across the ARTs

According to some of our interviewees, dependencies exist between requirements coming from these three ARTs. One ART might be the owner of a requirement and ask the other ART to implement this requirement for them. The current workaround to address this type of dependencies in the requirements documentation tool is to provide them as interfaces that are accessible by the other ART.

The other types of dependencies are in the form of requirements on a high level that are broken down and divided between the ARTs. So, each ART owns part of the higher level requirement. In that case, the artifacts related to different parts of the requirement will end up together in the integration process.

The following quote confirms the existence of dependency in requirements between the ARTs:

> *"they seem to be sharing the same test cases that relate to their requirements. Because, the test cases they're using are linked to requirements.[. . .] So they do have a dependency in requirements, but they don't have a common standard."*
>
> — A developer from the supporting ART

As it was mentioned by this developer, having alignment and standards is important due to these dependencies.

Dependencies between requirements from ART 1 and ART 2 and also ART 2 and ART 3 were mentioned during our interviews. Although, most of our interviewees from these three ARTs were not aware of the problems that exist related to these dependencies. Some of them stated that there are more dependencies between their ARTs and other ARTs that are working with hardware requirements. To address these dependencies and communicate these requirements a more aligned way between the ARTs can help them in the process of communicating their requirements.

The following quote emphasizes the need for standardization on the levels that these dependencies exist:

> *"I think it should be standardized on a higher level. So, it should be standardized on those requirements which then would interact with other ARTs. Because like I said, we have certain lower level requirements which need not be very informative to the other ARTs. So I would say that there we can have our own freedom and decide within our art as to how those lower level requirements should be written."*
>
> — A function developer

As it was mentioned by this function developer there is need for both customization and alignment in respect of the level of dependencies and the need for collaboration.

4.3.1.3 Common Guidelines Across the ARTs

As indicated before, employees complained about lack of awareness and understanding of how to work, when it comes to requirements and traceability. Again, not having a guidelines to follow is one of the reasons for this lack of awareness. Moreover, there is lack of shared understanding across the ARTs, due to not having a common guidelines / standards between them. While as it has been discussed before, there is a need for having a shared understanding across the ARTs, which can help with the process of integration of the trace links across the ARTs by the supporting ART.

Additionally, providing and maintaining separate guidelines for each ART is time consuming and requires resources. In fact, it is easier to provide and maintain a common guidelines for aligned processes and practices.

The following quote provides support for having a common way and guidelines across the ARTs for in-house software development:

> *"[...] there's no reason why we from ART 1 actually should have like different guidelines for the different levels. I think that would just be a good thing If we could have a common and standardized way of working on these different levels between ART 1 and ART 2 ."*
>
> — A system designer

As it was mentioned by this system designer, for the ARTs that are moving toward in-house software development, having a standard way and common guidelines for writing requirements could be beneficial.

4.3.1.4 Helps with Fulfilling Safety Standards

The company is required to fulfill the safety standards and every ART is responsible to meet these specific standards, while the current diverse workarounds across the three ARTs is affecting traceability and causing problems in the process of automating traceability. A more aligned and standardized way of working can facilitate meeting the safety standards across the ARTs. The following quote supports this reason:

> *"I think it's important that we standardize these things at our company, I mean, especially from a safety and legal point of view. It's good that we work in the same way."*

— A system architect

As it was mentioned by this system architect, having a standardized way of working is valuable from the safety perspective.

4.3.2 Scope of Alignment

For defining the scope of alignment, we mostly focused on alignment across the ARTs. But, we also asked about the employees opinion about alignment within the ARTs and across the company through our interviews and workshop.

Alignment within the ARTs has been challenging as our interviewees mentioned. But, looking inside the ARTs we can see that the requirements documentation practices are aligned to a high degree.

When asking about having alignment across the ARTs in the workshop, 69% of our participants stated their agreement and the other 31% responded "neutral", while surprisingly no one disagreed with the option of having alignment across the ARTs. The following quote from the interviews also supports the idea of alignment across the ARTs:

> *"[...] I think it should probably be across the ARTs, at least in our department. Yeah, I don't know if it's a good idea or if it's even possible to have a standard across the organization."*
>
> — A developer from the supporting ART

As it was noted in the above quote, having alignment across the ARTs working in the same department can be practical.

However, the responses were considerably different when asking about alignment on the company level. Only 32% of our respondents agreed with this option and 12.5% stated that they disagree with this option.

The following quote clarifies on how to define the scope for alignment:

> *"Well, at least if there is one team that should develop a framework for this. Then it's so much easier if everyone uses the same solution. Otherwise it's really hard to maintain. So, I would say at least for those ARTs that we support, I would like them to have the same way of working. Should it be the same across the company? But perhaps if they would like to use our solution? "*
>
> — A developer from the supporting ART

As it was mentioned by this developer from the supporting ART, alignment provides value when there is a need for a common solution.

4.3.3 Challenges for Alignment and Need for Customization

While alignment could bring advantages and help in many aspects as it was mentioned in the previous section, the process of moving toward alignment is challenging.

In fact, this would be challenging to make all these three ARTs following the same practices. Some of our participants mentioned that aligning practices of requirements documentation has been challenging even within their ART. But, our

findings shows that they have achieved a level of alignment in terms of abstraction level of requirements and to an extent in the type of provided contents in each level within the ARTs.

The following quote states that there is reluctance to change:

> *"Some ARTs maybe won't do that, and then they have their legacy, even if the structure was not good before, they go on with that because it takes so much time to rework everything. As I said, there has been a lot of resistance within my ART as well to do this."*
>
> — A system architect

As this system architect states due to legacy and the amount of work that is required for change, aligning is even challenging within one ART.

Every ART has its own way of working in dealing with requirements and traceability. Some of their practices have been established for a long time and changing them is challenging because it might require changes in many related parts. Changes in large corporations always necessitate a significant amount of time and effort, as well as the participation of individuals who are eager to change.

The following quote emphasizes the effect of alignment on different sections:

> *"I think it will pose a lot of problems because it will require the ARTs to probably change their way of working a lot and it will. It may also pose big challenges in modifying the requirements management system to accommodate the standard. And they have to kind of be willing to take on that work, depending on how much it is. To modify their data model. And in the requirements management system. "*
>
> — A developer from the supporting ART

As it was stated in the above quote, alignment requires change in different practices, the models and tools to be compatible with the new standards.

Apart from this as it was mentioned before, part of the differences are because of different needs that exist between the ARTs and even sometimes between the teams within one ART. Generally, having different needs can lead to a level of diversity that can be considered inevitable in practice. There is no tool at the moment that can address all the specific needs of different ARTs. For this reason, teams and ARTs would be left with no choice but adding the tools that they need to their existing processes. Additionally, not all the ARTs can use exactly the same requirement levels. Therefore, In order to fulfill the needs that are specific for a team or an ART, it is necessary to consider the need for diversity and customization in some parts.

4.4 Suggestions

As part of our study we asked for suggestions from our interviewees to help with the current problems related to diversity of requirements documentation and traceability practices. Further, these suggestions were validated through our workshop.

4.4.1 A more Mature Reference/ Standard Model

As it was mentioned before, there is a need for standards and guidelines that specifically address the requirements for in-house software development. Further, this reference/ standard model needs to be mature enough to address the needs of all the ARTs. In this section we go through the attributes that are needed to be in the model. Further, we discuss the need for having central initiatives to develop the model and the authority to maintain the model and be a reference point in terms of guidance and assessments for the model.

4.4.1.1 What to have in the model

S1. Requirement abstraction levels and trace links: Include all the different abstraction levels that are needed to fulfill the needs of these three ARTs. In addition, the types of trace links between requirements in different abstraction levels and a specific abstraction level with other types of artifact(e.g. code, test) are required to be described.

S2. Clear purpose of the content of requirement levels: The purpose of the content that is supposed to be provided for each level, needs to be addressed clearly. This can help to have a shared understanding of each level while writing a requirement in a specific abstraction level. Further, it can be used as a reference point for the supporting ART to know what is the type of content in each level. Moreover, this leads to reducing the inconsistencies that exist in the contents of the same abstraction levels.

S3. More examples for clarification: Supporting the model with more examples for how to break down requirements to different levels, what type of content to put on each level and how the trace links are supposed to be provided between the levels can mitigate the risk of misinterpretations by individuals.

S4. Avoid rigid constraints that dictate writing too much details: Having rigid constraints in a requirement abstraction level can cause going to a level of detail in its content that is not necessary for that level. In fact, this much detail might be required again in a lower abstraction level through the breakdown of that requirement. As a result, we can end up having the same content or duplicated information in two different levels of a specific requirement.

Based on the survey results (Figure 4.5), most participants also acknowledged that a standard model should include the parameters we outlined in Section 4.4.1.1.

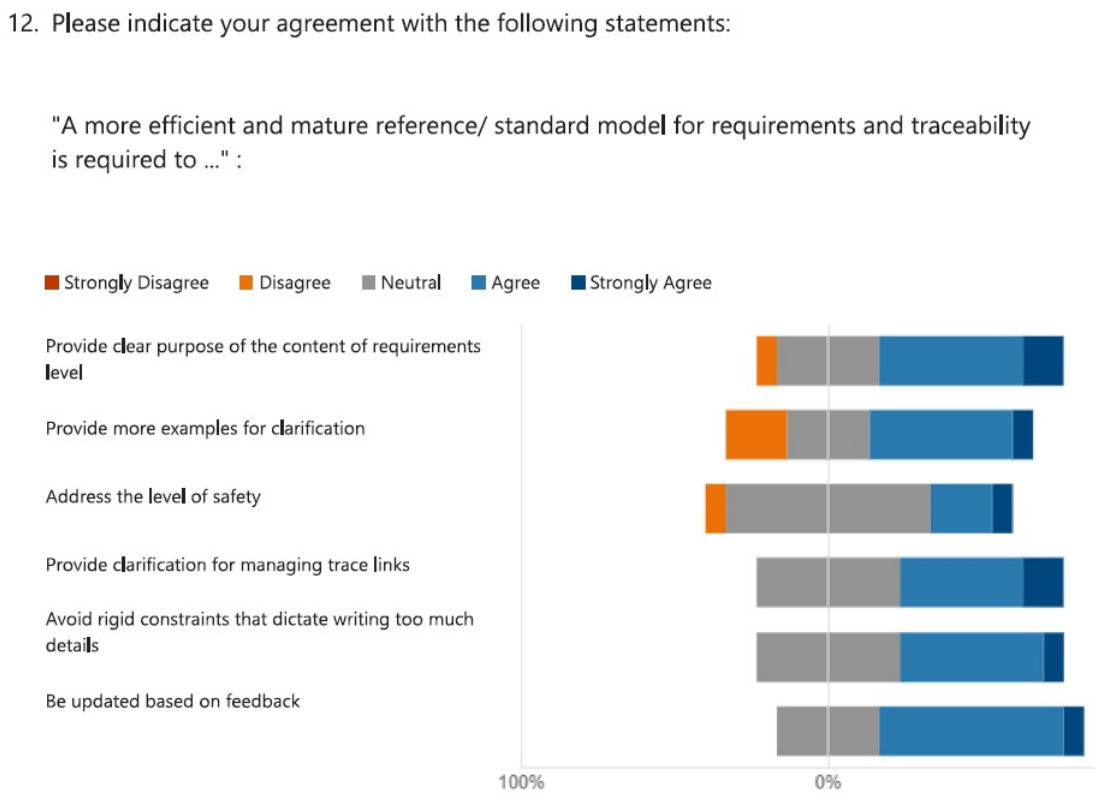


Figure 4.5: Survey results for the reference/ standard model

4.4.1.2 Central initiatives to develop the model

For the reference/ standard model to provide value in practice, it is required to be managed by an authority. The authority is supposed to have some responsibilities related to managing the model:

S5. These central initiatives can be formed based on discussion and agreement across the ARTs. Representatives from the three ARTs and the supporting ART can come together and discuss the current workarounds and share their problems and needs together. Further, they can share their knowledge about different aspects of their requirement and traceability practices and use this knowledge as a base for developing this model.
The following quote indicates that central initiatives should be based on discussion across the ARTs:

> *"[...]rather than someone from up there telling us you should do like this and I don't understand it and others I talked to don't understand it and if I see someone else is not really working according to that, then it's not no point for me there, so it's. Yeah, having a more kind of agreement between all ARTs would help a lot, I would say".*
>
> — A function owner

As it was stated by this function owner, engagement of the employees who work directly on requirements can provide more value for reaching a common way of working.

S6. Another suggestion is that the model be proposed from a higher level in the department by a group with vision about the practices used by these three ARTs. In that case, the ARTs can provide their feedback on this model and revisions can be applied on a model based on the feedback from the ARTs.

The following quote supports the idea of providing a proposal from a higher level in the department to develop the model:

> *"[...] if there is a proposal that is presented to our ART like we think everyone should work like this now and we can review it, then I think it's fine. But like going into discussions with all the other ARTs to try to find that way, I would not be part of that."*
>
> — A product owner

As it was stated by this product owner from one of the ARTs, this way would be more efficient than going through discussion with other employees across the ARTs.

4.4.1.3 An Authority to Manage the Model

For the reference/ standard model to provide value in practice, it is required to be managed by an authority. The authority is supposed has some important responsibilities in managing the model:

S7. Coaching the teams to follow the reference model: The authority is expected to coach the teams and assist them in following the reference model. It should be noted that for the reference model to be followed by the teams, it is required to be supported by an authority to be there for addressing ambiguities and providing clarifications on the model.

S8. Provide assessment on the teams work to be compatible by the model: Assessment on the output provided by teams are needed based on the model. This can help the teams to improve their work based on the assessments from the authority. Additionally, it can identify the mistakes made by the individuals, which can be based on lack of knowledge or just unavoidable human mistakes.

S9. Update the model based on feedback: For the model to be efficient and cope with the process evolution and unexpected needs, it is essential to be updated based on changes in the ARTs' needs. The authority is responsible for having a solution to get frequent feedback on the model. Furthermore, the authority is expected to have a plan to improve the model based on the feedback received from the ARTs.

Figure 4.6 displays a table that consists of the interviewees suggestions about having a reference model across the ARTs with several specifications and it can be used as a suggestion that can contribute to mitigate the effect of the factors that cause differences in requirements documentation and traceability practices across the ARTs. This table illustrates clearly the relation between these suggestions and problems based on the study results.

Figure 4.6: Suggestions that can contribute to mitigate the effect of the factors leading to diversity of practices across the ARTs

4.4.2 Managing tools for requirements and traceability

As it was discussed before, some of the reasons for the problems of integrating traceability across the ARTs are related to the tools that are somehow connected to requirements. The following suggestions are related to tools:

S10. More stable tool before use: Having a more stable tool before starting to use it through the organization can help with avoiding unexpected diversities, which can happen due to frequent changes. Additionally, for the official requirement tools it is better that they come with all the required features related to traceability before putting them into use. This is required to avoid the need to add other tools through the process by different teams.

S11. Do all the breakdown of requirements in one tool: By having just one tool that consists of all the data of the breakdown of requirements, we can have all the trace links between different requirement levels in one tool. This facilitates the process of integrating traceability across the ARTs by providing more consistency and decreasing the need for integration between different tools.

S12. Keep the implementation level requirements in Simulink: In order to avoid having implementation level requirements in the official requirement tool and giving more freedom to the developers, we can keep these types of requirements in Simulink. Indeed, by following this way the supporting ART will have to provide an extra solution to extract the requirements from Simulink. Further, another solution is required to integrate Simulink with the official requirement tool to avoid the current solution of adding the requirement id in Simulink.

In the table below (Figure 4.7) we can see the relation between the interviewees suggestions that helps to manage tools across the ARTs.

Solutions / *Reasons Of Problems*			Suggestions Related To Managing Tools		
			More stable tool before use	Do all the breakdown of requirements in one tool	Keep the implementation level requirements in Simulink
			S10	S11	S12
R1: Tools	Tool transitioning	R1.1	✓		
	A new evolving tool	R1.2	✓		
	Different tools for different need	R1.3	✓	✓	✓

Figure 4.7: Suggestions for managing tools across the ARTs

5
Discussion

In this chapter we will go through the discussion of the findings of our study. First, in section 5.1 , the relation between the findings and the research questions is provided. Then, in section 5.2, a comparison of our study to related works will be discussed. In section 5.3 , we will talk about how our findings can be valuable in the case company and similar cases in practice. Finally, in section 5.4, the threats to validity and what we did in order to decrease their impact or mitigate them.

5.1 Findings in Relation to Research Questions

According to our results (Section 4.1), inconsistencies and missing data in requirements and trace links lead to problems for automating traceability integration.

Section 4.1.1, answers our first research question (RQ1) by identifying the differences that lead to inconsistencies in requirements and their corresponding trace links, which adversely affect collaboration across the ARTs with the aim of integrating traceability. Differences were found in the tools being used for managing requirements, requirement abstraction levels, type of contents in each level and traceability practices(the workarounds to link requirements to other requirements or requirements to other artifacts (e.g. implementation, test)).

Apart from this, Section 4.1.1 gives us some insight for our third research question (RQ3) by explaining how these differences are contributing to inconsistencies that are causing delays in the process of automating the traceability integration across the ARTs.

Additionally, problems other than diversity of requirements documentation and traceability practices that are affecting the integration of traceability are explained in the results (Section 4.1.2). Missing data related to requirements and trace links is a problem that exists within the ARTs and can be categorized as a traceability problem in smaller units like teams, which in the end contributes to the main problem of integrating traceability across the ARTs.

Furthermore, we found the reasons behind the differences in requirements documentation and traceability practices (Section 4.2). This addresses our second research question (RQ 2). These reasons cover all the identified differences in Section 4.1.1. These reasons are categorized as factors that cause challenges related to tools, guidelines, applying large-scale agile and prioritizing traceability. Apart from this, having different needs was identified as a reason that justifies the need for traceability.

In Section 4.3, we can see the reasons that support the idea of moving toward more alignment across the ARTs. This section provides further information to answer our third research question (RQ 3).

The results (Section 4.3) indicate that more consistency in requirements and trace links is one of the outcomes of aligning requirements documentation and traceability practices, which can help with one of the main challenges in our study (Section 4.1.1). Furthermore, a shared understanding is required between the ARTs and the supporting ART to be able to provide solutions for extracting the data for traceability integration. Due to the dependencies that exist between the ARTs, a shared understanding would provide more ease in communication in terms of dependent requirements. Reaching to this shared understanding has been a great challenge due to the diverse practices across the ARTs. More aligned requirements documentation and traceability practices supported by a common guideline across the ARTs can promote this shared understanding. Finally, the results show that alignment brings benefits of a more standardized way, which helps with fulfilling safety standards.

In addition, more aligned practices across the ARTs, gives the opportunity for a more common solution for extracting the data and integrating traceability. Whereas, the current existing diversity between the ARTs requires maintaining different solutions, which is not sustainable considering that more alignment is applicable to an extent.

Otherwise, the ARTs need to provide the data themselves in a format that is understandable and acceptable by the supporting ART. In fact, making the ARTs responsible for providing this data means asking them to provide their own solutions to collect the data and present the data in a common format.

Moreover, our findings (Section 4.3) suggest that the need for customization should be considered, while talking about alignment. As it is elaborated in Section 4.2, differences in needs across the ARTs justify the necessity of customization in some parts. The need for using different abstraction levels across the ARTs and the demand by some teams for specific features that can not be addressed by the requirement tool should be taken into account.

In the Section 4.4, suggestions are provided in terms of moving toward more alignment, while considering the need for customization. This section focuses on our third research question (RQ 3).

Our findings (Section 4.4.1) suggest that having a reference/ standard model for requirements and trace links related to requirements across the ARTs, can provide solutions for the problems arising from diversities. The model can be used as a reference point across the ARTs to promote a shared understanding of the abstraction levels, type of the content that needs to be provided in each level and the required trace links related to these levels. The model can be used to decrease the inconsistencies related to misinterpretations, by clarifying the purpose of each abstraction level and further lead to a more aligned approach in writing requirements. Moverover, some of the abstraction levels are common across the ARTs , while some of the levels are connected to the specific needs within the ARTs (Section 4.1.1.2). The model is required to address all these different levels to provide customization based on the specific needs in the ARTs and alignment where there are common needs.

In order for the reference/ standard model to keep pace with the changes in the process of the ARTs, there is a need for an authority to manage the model. More simply, the authority should take the responsibility to update the model based on the feedback from the ARTs. Otherwise, as it was mentioned in Section 4.2.2.2, deviations will happen as time passes due to the model not being able to fulfill the

ARTs' needs and it will cause the model to fail its objectives. Further, assessments on the requirements and their tracelinks can be applied in reference to this model.

5.2 Implications for Research

In this section we go through a comparison of our findings with the other results from the related works to our case study.

Wohlrab et al. [5] suggests why there is a need for diversity in RE practices and how diversity and alignment of RE practices is addressed in corresponding RIMs, while in our study we focus on the factors that contribute to the existing diversity in requirements documentation and traceability practices (Section 4.2), whether it's needed or is an outcome of another problem. Our findings indicate that some of the reasons for the differences in requirement documentation and traceability practices, across the ARTs that are working in the same department, are the outcomes of problems related to tools, guidelines, applying large scale agile and priority of traceability. However, our findings also confirm the need for customization of requirements levels and related tools. Further, the suggestions in our study for the reference model is focused on managing requirements structure and trace of links, while the RIMs in their work focuses on a broader range in RE concepts. But, the need for diversity and customization is confirmed in both of these studies.

Maro et al. [4] mention diversity in tools and artifacts as one of their identified challenges for traceability in automotive companies. Further, they suggest that integration is required to be applied between different tools. While diverse tools in their study refers to the different tools that are used through the whole process of development, we explored differences in tools in a more limited scope which is focused on tools for managing requirements. Our study also confirms that diverse tools lead to traceability problems (Section 4.1.1.1) and integration between tools is essential. Additionally, our study extends their findings by exploring what contributes to the use of diverse tools. Our findings (Section 4.2.1) indicate that through the transition process from an old tool to a new one the teams might face the problem of dependency on the old tool. Moreover, it shows that incompleteness and lack of stability in a new tool that is still evolving can justify the need to add other tools to the process, which finally leads to more diversity. Furthermore, in Section 4.4.2, it is suggested to start using the tool after it is more stable to avoid the associated problems that lead to diversity.

In another study by Karhapää et al. [17], several challenges have been diagnosed for reaching alignment. They describe that utilizing diverse tools is common in order to fulfill different needs, which raises problems when it comes to integration and alignment. Our study also confirms that the diversity of tools that are being used across the ARTs is partly affected by different needs (Section 4.2.5). Further, we described the use of different tools as one of the main challenges in integrating traceability. Additionally, we explored the factors that lead to the diversity in tools (Section 4.2.1). Apart from this, they state that missing links between requirements and other artifacts is a great issue in providing traceability. Moreover, they add that one of the causes of these missing links can be legacy, an example of this is the unspecified relationship of test cases with requirements. We also identified missing links as part of the missing data related to requirements and traceability in our study to be a challenge for integration (Section 4.1.2). In addition, we found missing data

in other parts. For instance, the data related to the status of requirements are not complete and also due to the lack of version control in trace links, tracking changes related to the history of trace links is challenging.

Dömges et al. [29] state that lacking a structured way of tracing adversely affects the quality of the software, as well as negatively impacts the organization because of dependency on individual knowledge and lack of common understanding of the tracing process. Moreover, Pinheiro [21] adds that in complex systems, setting an environment that is based on a traceability model, consisting of tools and procedures with the support of automated approaches is needed, so that it can handle the complications based on the extensive amount of data. In Section 4.1.1.3 we described that there are differences in traceability practices due to not having an official workaround to trace requirements even within one ART. Additionally, the current approaches to provide traceability are manual. Furthermore, the diverse manual workarounds to address traceability contribute to inconsistencies and problems in collaboration across the ARTs for integration.

Neto et al. [38] mention that collaboration is essential across the boundaries of the organization and lack of cooperation on this level can lead to more diversity in their practices and workarounds. Our study also confirms lack of communication and collaboration across the ARTs as one of the factors that causes differences across the ARTs.

Sikora et al. [28] states that guidelines supporting reference models for specifying requirements at different levels of abstraction are required to address the content of each level and the relation between different levels. Using a reference model was also suggested in our study to mitigate the risk of misinterpretations. Further, our findings suggest that in order for this suggestion to be applicable in practice, avoiding rigid constraints in providing guides for how to write the content of levels can be helpful. Otherwise, this could lead to writing unwanted details in a specific level that can cause duplication in different levels of the same requirement.

5.3 Implications for Practice

Our findings show that there is lack of guidance and standards in managing requirements and traceability for in-house software development in the case company. This is affecting the quality of the data that is provided for requirements and the trace links related to requirements. In fact, the provided data is not compatible across the ARTs, which has led to the need for maintaining different solutions for each ART for automatic data extraction.

Further, maintenance of different solutions in practice leads to an increased amount of work and establishing manual solutions is much more expensive than it has to be. Therefore, to be able to extract the data automatically from the ARTs one solution is to increase consistency of the data across the ARTs.

Our study provided suggestions in order to move toward more alignment which can further lead to more compatibility of the requirements and traceability data across the ARTs. These suggestions are provided in Section 4.4 to mitigate the problems related to guidelines and tools that can lead to deviations and diversities in requirements documentation and traceability practices.

It should be mentioned that the main focus of our study was to identify the factors leading to diversity, which could be common in other similar cases. So, our

suggestion is to look more deeply into these factors and provide solutions to mitigate them. We believe that there is room for increasing alignment considering the fact that some of the factors can be looked at as a problem that could be provided with solutions. More stability and completeness is required before adding tools or practices to a process in companies that are consisting of different units which are having dependency on each other. This is due to the fact that the way of handling changes and timings can differ in different teams which can lead to diversity in a short period of time across these units. Moreover, the problem of lack of awareness about traceability importance is required to be addressed in the companies, specially when it comes to safety critical domains. Finally, in the case of applying agile on a large scale, the effect of freedom in choosing practices and the need for standards is required to be taken into account.

5.4 Limitations and Delimitations

The data in our study is collected from different sources, which has given us the opportunity to triangulate and to combine the interpretations provided based on each source [11]. So, the impact of each interpretation has become limited and the conclusion is based on the analysis on interpretation from these different sources (triangulation) [11].

Researcher bias is also one factor that should be taken into consideration when it comes to interpreting qualitative data [11]. But since we had more than one researcher in our study, the results of data analysis is based on the combination of different interpretations which increases the validity of our results [11].

Different aspects of validity [11, 13] for this case study should be considered as follows:

- *Construct validity:* In conducting interviews and surveys, we considered the fact that there might be misunderstanding of concepts and questions [11]. This was due to the fact that the ARTs might have their own conventions and way of interpreting different concepts. Therefore, we provided definitions for the main concepts in the process of designing our interview guide.
  Further, for the workshop and survey, where representatives were gathered from the three ARTs and the supporting ART, we decided to divide the survey in several sections and give a more detailed explanation of each section during the workshop then ask our participants to provide answers to the survey questions based on our explanations. Moreover, because it was important for us to validate the results of the study through surveys, providing accurate definitions of the interview results mitigated the possibility of misunderstanding during the workshop.
- *External validity:* The results of this study is based on the data provided by different ARTs in one company. One aspect that is related to external validity is how the results from this study can be relevant to cases with common attributes and how it can be generalized for other cases [11]. Based on the fact that our sample is limited to one company, we tried to increase diversity among the company representatives as much as possible.
  We tried to invite representatives from different roles and teams of each ART to the interviews. Moreover, for attending the workshop and the surveys, in

addition to previous interviews' participants, we invited other employees from the three ARTs and the supporting ART.
As a result, we received 5 extra survey responses from the workshop participants who had not attended the interview sessions before.
Further, a suggestion to increase generalizability is that in a future research, a similar survey could potentially be conducted with other companies to investigate generalizability beyond the case company, but it was not of priority for our thesis.

- *Reliability:* One expectation that matters here is that if other researchers carried out this study in the future, they should get the same findings. While this ideal goal can be difficult to achieve in qualitative studies, we aimed to at least provide recoverability, i.e. allow other researchers to achieve deep understanding on why their insights might differ from ours. In order to support this goal, we documented the process of our study in sufficient detail. The instructions of data analysis and coding (Section 3.3), the questions in interviews (Appendix A) and survey (Appendix B) and also the ways that we categorized data (Appendix G) is clearly documented. Further, it can be investigated whether an overview of themes, codes, and example quotes can be publicly shared.

6
Conclusion

This study aimed to find the differences in requirements documentation and traceability practices that are adversely affecting collaboration across the ARTs to develop and maintain solutions for traceability integration and further the reasons behind these differences. By analyzing the reasons behind the diverse practices across the ARTs, we came to the conclusion that there are different factors involved in leading to diversity in practices.

Our findings indicate that diversity is unavoidable in order to provide different teams/ ARTs with the means they require to achieve their needs. This is partly due to the fact that there are different needs across the ARTs and even within one ART that are required to be addressed.

However, the factors that lead to differences in requirements documentation and traceability practices are not limited to the conceptual differences of needs that validate the necessity for having diversity. Our study shows that diversity and deviations are partly related to problems in guidelines and tools that are required to facilitate the way of working for the employees. These problems are leaving employees with no alternatives other than finding their own solutions or deviating from the defined processes. Due to the participants of our study these deviations that lead to diversities are not always desired but in many cases are the only option to make progress in their work.

Moreover, based on our analysis, diversity in requirements documentation and traceability practices is identified as one of the main challenges for automating the process of integrating traceability across the ARTs. This is due to the fact that this diversity exists to a degree that has introduced considerable inconsistencies in the required data for integration. In fact, this is leading to the need to provide and maintain different solutions for the ARTs in order to extract the essential data for integration. This should be mentioned that maintaining different solutions demands resources.

Furthermore, our findings show that there are other priorities (e.g. deliveries) that will affect the availability of resources for traceability related tasks. So, making the processes related to traceability more sustainable is a necessity. Therefore, finding solutions to decrease the level of diversity and provide more consistency in the input required for tracing requirements is of great value.

The participants in our study suggested that a reference/ standard model across the ARTs for requirements and their corresponding trace links can facilitate their work by providing more clarification along with promoting alignment. Further, maintaining the reference model based on the process evolution and feedback from the users along with applying assessments on the data, provided by the teams, based on the model is of high importance. Moreover, managing tools in terms of providing more stability and completeness has been proposed to avoid undesired diversity.

In addition, our findings confirm that integration between different tools is of high importance to promote consistency in the requirements and traceability data.

Bibliography

[1] Broy, M. (2006). Challenges in Automotive Software Engineering. Proceedings of the 28th International Conference on Software Engineering. https://doi.org/10.1145/1134285.1134292

[2] Wohlrab, R., Pelliccione, P., Knauss, E., Larsson, M. (2019). Boundary Objects and their use in Agile Systems Engineering. Journal of Software: Evolution and Process, 31(5). https://doi.org/10.1002/smr.2166

[3] Barlow, J. B., Giboney, J. S., Keith, M. J., Wilson, D. W., Schuetzler, R. M., Lowry, P. B., Vance, A. (2011). Overview and guidance on agile development in large organizations. Communications of the Association for Information Systems, 29, 25–44. https://doi.org/10.17705/1cais.02902

[4] Maro, S., Steghöfer, J.-P., Staron, M. (2018). Software traceability in the automotive domain: Challenges and solutions. Journal of Systems and Software, 141, 85–110. https://doi.org/10.1016/j.jss.2018.03.060

[5] Wohlrab, R., Knauss, E., Pelliccione, P. (2020). Why and how to balance alignment and diversity of requirements engineering practices in Automotive. Journal of Systems and Software, 162, 110516. https://doi.org/10.1016/j.jss.2019.110516

[6] Ebert, C., Favaro, J. (2017). Automotive software. IEEE Software, 34(3), 33–39. https://doi.org/10.1109/ms.2017.82

[7] Leffingwell, D. (2010). Agile software requirements lean requirements practices for teams, programs, and the enterprise. Addison-Wesley Professional.

[8] Wohlrab, R., Steghofer, J.-P., Knauss, E., Maro, S., Anjorin, A. (2016). Collaborative Traceability Management: Challenges and opportunities. 2016 IEEE 24th International Requirements Engineering Conference (RE), 216–225. https://doi.org/10.1109/re.2016.17

[9] Dybå, T., Dingsøyr, T. (2008). Empirical studies of Agile Software Development: A systematic review. Information and Software Technology, 50(9-10), 833–859. https://doi.org/10.1016/j.infsof.2008.01.006

[10] Fucci, D., Palomares, C., Franch, X., Costal, D., Raatikainen, M., Stettinger, M., Kurtanovic, Z., Kojo, T., Koenig, L., Falkner, A., Schenner, G., Brasca, F., Männistö, T., Felfernig, A., Maalej, W. (2018). Needs and challenges for a platform to support large-scale requirements engineering. Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement. https://doi.org/10.1145/3239235.3240498

[11] Runeson, P., Höst, M. (2008). Guidelines for conducting and reporting case study research in software engineering. Empirical Software Engineering, 14(2), 131–164. https://doi.org/10.1007/s10664-008-9102-8

[12] Stol, K.-J., Fitzgerald, B. (2018). The ABC of Software Engineering Research. ACM Transactions on Software Engineering and Methodology, 27(3), 1–51. https://doi.org/10.1145/3241743

[13] Yin, R. K. (2003). Case study research: Design and methods (3rd ed.). Sage.

[14] Saldaña,J. (2013). The coding manual for qualitative researchers (2nd ed.) . Sage.

[15] Robson, C. (2002). Real World Research (2nd ed.). Blackwell.

[16] Wohlrab, R., Pelliccione, P., Knauss, E., Gregory, S. C. (2018). The problem of consolidating re practices at Scale: An ethnographic study. Requirements Engineering: Foundation for Software Quality, 155–170. https://doi.org/10.1007/978-3-319-77243-1,10

[17] Karhapää, P., Haghighatkhah, A., Oivo, M. (2017). What do we know about alignment of requirements engineering and software testing? Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering. https://doi.org/10.1145/3084226.3084265

[18] Pinheiro, F.A.C. (2004). Requirements Traceability. In: do Prado Leite, J.C.S., Doorn, J.H. (eds) Perspectives on Software Requirements. The Springer International Series in Engineering and Computer Science, vol 753. Springer, Boston, MA.

[19] Hong, Y., Kim, M., Lee, S. W. (2010, May). Requirements management tool with evolving traceability for heterogeneous artifacts in the entire life cycle. In 2010 Eighth ACIS International Conference on Software Engineering Research, Management and Applications (pp. 248-255). IEEE.

[20] Peraldi-Frati, M. A., Albinet, A. (2010, April). Requirement traceability in safety critical systems. In Proceedings of the 1st Workshop on Critical Automotive applications: Robustness and Safety (pp. 11-14).

[21] Pinheiro, F. A., Goguen, J. A. (1996). An object-oriented tool for tracing requirements. IEEE software, 13(2), 52-64.

[22] Rempel, P., Mäder, P. (2016). Preventing defects: The impact of requirements traceability completeness on software quality. IEEE Transactions on Software Engineering, 43(8), 777-797.

[23] Kasauli, R., Knauss, E., Horkoff, J., Liebel, G., de Oliveira Neto, F. G. (2021). Requirements engineering challenges and practices in large-scale agile system development. Journal of Systems and Software, 172, 110851. https://doi.org/10.1016/j.jss.2020.110851

[24] Dikert, K., Paasivaara, M., Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations: A systematic literature review. Journal of Systems and Software, 119, 87–108. https://doi.org/10.1016/j.jss.2016.06.013

[25] Grimm, K. (2003, May). Software technology in an automotive company-major challenges. In 25th International Conference on Software Engineering, 2003. Proceedings. (pp. 498-503). IEEE.

[26] Luyet, V., Schlaepfer, R., Parlange, M. B., Buttler, A. (2012). A framework to implement stakeholder participation in environmental projects. Journal of environmental management, 111, 213-219.

[27] Eliasson, U., Heldal, R., Knauss, E., Pelliccione, P.: The need of complementing plan-driven requirements engineering with emerging communication: experiences from Volvo Car Group. In: RE 2015, pp. 372–381. IEEE, August 2015

[28] Sikora, E., Tenbergen, B., Pohl, K. (2011, March). Requirements engineering for embedded systems: An investigation of industry needs. In International Working Conference on Requirements Engineering: Foundation for Software Quality (pp. 151-165). Springer, Berlin, Heidelberg.

[29] Dömges, R., Pohl, K. (1998). Adapting traceability environments to project-specific needs. Communications of the ACM, 41(12), 54-62.

[30] Heindl, M., Biffl, S. (2005, September). A case study on value-based requirements tracing. In Proceedings of the 10th European software engineering conference held jointly with 13th ACM SIGSOFT international symposium on Foundations of software engineering (pp. 60-69).

[31] Liebel, G., Tichy, M., Knauss, E., Ljungkrantz, O., Stieglbauer, G. (2018). Organisation and communication problems in automotive requirements engineering. Requirements Engineering, 23(1), 145-167.

[32] Attarha, M., Modiri, N. (2011, August). Focusing on the importance and the role of requirement engineering. In The 4th International Conference on Interaction Sciences (pp. 181-184). IEEE.

[33] Lester, J. N., Cho, Y., Lochmiller, C. R. (2020). Learning to do qualitative data analysis: A starting point. Human Resource Development Review, 19(1), 94-106.

[34] Elliott, V. (2018). Thinking about the coding process in qualitative data analysis. The Qualitative Report, 23(11), 2850-2861.

[35] Linneberg, M. S., Korsgaard, S. (2019). Coding qualitative data: A synthesis guiding the novice. Qualitative research journal.

[36] Benbasat, I., Goldstein, D. K., Mead, M. (1987). The case research strategy in studies of information systems. MIS quarterly, 369-386. MIS Q 11(3):369–386 doi:10.2307/248684

[37] Stake, R. E. (1995). The art of case study research. sage.

[38] Neto, F. G. D. O., Horkoff, J., Knauss, E., Kasauli, R., Liebel, G. (2017, September). Challenges of aligning requirements engineering and system testing in large-scale agile: A multiple case study. In 2017 IEEE 25th International Requirements Engineering Conference Workshops (REW) (pp. 315-322). IEEE.

A

Appendix 1

A.1 Interview Guide

A.1.1 Introduction

1. Introduce researchers.
2. Outline the interview objectives:
   - The interview will be conducted in order to collect information related to the case study, which investigates Traceability and Requirements Documentation Practices across the ARTs.
3. Ask interviewee about the consent:
   - The interview will be recorded with your consent, using the video recording and live caption option for generating transcripts in Microsoft Teams.
   - Any identifiable information of the interviewee will not be used during analysis of the transcripts and the future results that will be published of our study.
   - The anonymous interview transcripts will be accessible to the academic supervisor, industry supervisor and the interviewers.
   - The recorded videos and generated transcripts in Microsoft Teams will be deleted by the interviewers after the Master thesis is completed.
   - It is possible to skip some questions during the interview for any reason from the interviewee or interviewers.

A.1.2 Interview Questions

Note: In several cases, the blue color questions were asked instead of the black color questions from the supporting ART interviewees.

General:

1. Which ART are you working in? Which team?
2. Tell us about your role in the ART?
3. How many years of experience do you have working related to Requirements or Traceability?
4. Could you give a short description of the functional area of your ART?
5. How many teams/ARTs do you collaborate with in your daily work?

Requirement Documentation Practices:

Before ask the following questions, we explain:

- What we mean by other ARTs

- The definition of Requirement Documentation Practices which include Tools, Requirements structure, Requirements Styles,Types of Trace link, Types of Generating reports for the supporting ART

1. Could you tell us about the Requirement Documentation Practices being used:
   (a) In your team? In other ARTs?
   (b) What are the similarities and differences with other teams in your ART?Don't ask from the supporting ART
   (c) What are the similarities and differences with other ARTs? Similarities and differences between ARTs
2. When did you start using these practices? Do you know how long these practices have been used?
3. Are these practices modified at any point?
   (a) What were the drivers?
4. Could you explain the problems related to current Requirements Documentation Practices?
   (a) Any plans for modifying these practices in order to solve these problems
   (b) Do you have any suggestions related to that?
5. To what extent Requirements Documentation Practices are standardized within your company?
   (a) To what degree these standards are being followed?
   (b) To what extent you need to customize these practices and define your own solutions? To what extent these practices are customized in the ARTs?
   (c) How do you see the results of having standardized Requirements Documentation Practices across the ARTs?
6. What are the reasons behind differences of Requirement Documentation Practices?
   (a) Between teams working in the same ART.
   (b) Between different ARTs.
7. What are the problems derived from these differences of Requirement Documentation Practices?
8. To what extent do you see dependency between requirements from different ARTs?
   (a) What is current solution or workaround to address these dependencies?
   (b) Which problems derived from the current solutions?
   (c) What is your suggestion to tackle these problems?

Traceability:

1. Which trace links related to Requirements Documentation Practices are required to be applied?
2. To what extent the current Requirement Documentation Practices have caused problems with providing these trace links?
3. What are your current practices in order to facilitate traceability related to requirements ?
   (a) Is it sustainable or needs some changes?

(b) Any plans or suggestions to improve?

Lifecycle:

Before ask the following questions, we explain the definition of requirement lifecycle.

1. Could you explain about the requirement lifecycle in your ART/Team? Could you explain about the requirement lifecycle in other ARTs?
   (a) Any instruction or predefined standards?
   (b) Interaction with suppliers?

Large-Scale Agile:

1. How do you see the transition to Large-Scale Agile and the changes in ways of collaboration, causing problems related to Requirement Documentation Practices?

A.1.3 Additional comments

We appreciate your participation in this interview and the information you provided is valuable for our study.
Is there anything else you would like to add?

B

Appendix 2

B.1 Survey Questions

Survey-Msc-Thesis-Traceability and Diversity of Requirements Documentation Practices

1. Have you participated in one of the interviews related to our study?

- ◯ Yes
- ◯ No

2. Please select your ART:

- ◯ ART1
- ◯ ART2
- ◯ ART3
- ◯ Supporting ART

3. Please state if you disagree with anything that we mentioned about the **"Requirement Levels"** or if there is something else you want to add?

4. Please state if you disagree with anything that we mentioned about **"tools"** or if there is something else you want to add?

5. Please state if you disagree with anything that we mentioned about **"traceability tools/ practices"** or if there is something else you want to add?

6. Please indicate your agreement with the following statements:

"One of the reasons behind differences in requirements documentation and traceability practices between ARTs is ...":

	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
No clear guideline/ standard for in-house development	◯	◯	◯	◯	◯
Tools	◯	◯	◯	◯	◯
Lack of authority	◯	◯	◯	◯	◯
Freedom in teams to decide on their own way of working	◯	◯	◯	◯	◯
Lack of communication / collaboration across ARTs	◯	◯	◯	◯	◯
Different needs on tools/ requirement levels	◯	◯	◯	◯	◯
Priority of traceability	◯	◯	◯	◯	◯

7. Please indicate your agreement with the following statements:

"Moving toward more alignment is required because of ..."

	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Automation and time efficiency	◯	◯	◯	◯	◯
Dependency across ARTs	◯	◯	◯	◯	◯
Providing ease and generic guideline for clarification	◯	◯	◯	◯	◯
Fulfilling safety standards	◯	◯	◯	◯	◯
More consistency in requirements and trace links	◯	◯	◯	◯	◯
Common/ standardized way of working can be beneficial	◯	◯	◯	◯	◯

8. Please indicate your agreement with the following statements:

"To help with solving the problems mentioned related to differences of requirements documentation and traceability practices , a more common way of working/ standard needs to be applied ..."

	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Within ART	◯	◯	◯	◯	◯
Across the ARTs	◯	◯	◯	◯	◯
On the company level	◯	◯	◯	◯	◯

9. Please indicate your agreement with the following statements:

"One of the important **challenges** for moving toward alignment is ..."

	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
It's hard to change ARTs current practices	◯	◯	◯	◯	◯
It takes time	◯	◯	◯	◯	◯
There is always need for customizatio n in some parts based on different needs	◯	◯	◯	◯	◯

10. Please indicate your agreement with the following statements:

"In order to solve the problems mentioned, related to tools, we need to ..."

	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Have all the break down of the requirements in the new tool	◯	◯	◯	◯	◯
Update the new tool to better support traceability	◯	◯	◯	◯	◯
Making the tools more stable before start using(e.g. the new tool)	◯	◯	◯	◯	◯
Use Simulink for low level, the new tool for high level requirements and provide integration between them	◯	◯	◯	◯	◯
A tool that integrates wit h development process and keep track of requirement s tatus	◯	◯	◯	◯	◯

11. How do you think a more mature reference/ standard model for requirements and traceability can be effective in solving problems related to differences in requirement documentation and traceability practices between ARTs?

Not Effective	Slightly Effective	Moderately Effective	Very Effective	Extremely Effective
◯	◯	◯	◯	◯

12. Please indicate your agreement with the following statements:

"A more efficient and mature reference/ standard model for requirements and traceability is required to ..." :

	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Provide clear purpose of the content of requirements level	◯	◯	◯	◯	◯
Provide more examples for clarification	◯	◯	◯	◯	◯
Address the level of safety	◯	◯	◯	◯	◯
Provide clarification for managing trace links	◯	◯	◯	◯	◯
Avoid rigid constraints that dictate writing too much details	◯	◯	◯	◯	◯
Be updated based on feedback	◯	◯	◯	◯	◯

13. Please indicate your agreement with the following statements:

"A central initiative for developing a reference/ standard model for requirements and traceability is better to be based on ..."

	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree
Discussion and agreement across ARTs	◯	◯	◯	◯	◯
A proposal from a higher level and then revisions on model based on feedback from ARTs	◯	◯	◯	◯	◯

14. Do you have any other suggestion to solve the problems related to differences across ARTs?

C

Appendix 3

C.1 Survey Response

Note: In order to keep anonymity at the survey response, the name of the company's ARTs and some of the company tools are covered.

Survey-Msc-Thesis-Traceability and Diversity of Requirements Documentation Practices

16	33:19	Active
Responses	Average time to complete	Status

1. Have you participated in one of the interviews related to our study?

Yes	11
No	5

2. Please select your ART:

[redacted] ART 1	4
[redacted] ART 2	5
[redacted] ART 3	2
[redacted] Supporting ART	5

3. Please state if you disagree with anything that we mentioned about the **"Requirement Levels"** or if there is something else you want to add?

12
Responses

Latest Responses

"Disagree [redacted] not used for requirement writing in [redacted] i..."

"First of all, really nice work! And wow, who can say we don't hav..."

5 respondents (**42**%) answered **levels** for this question.

4. Please state if you disagree with anything that we mentioned about **"tools"** or if there is something else you want to add?

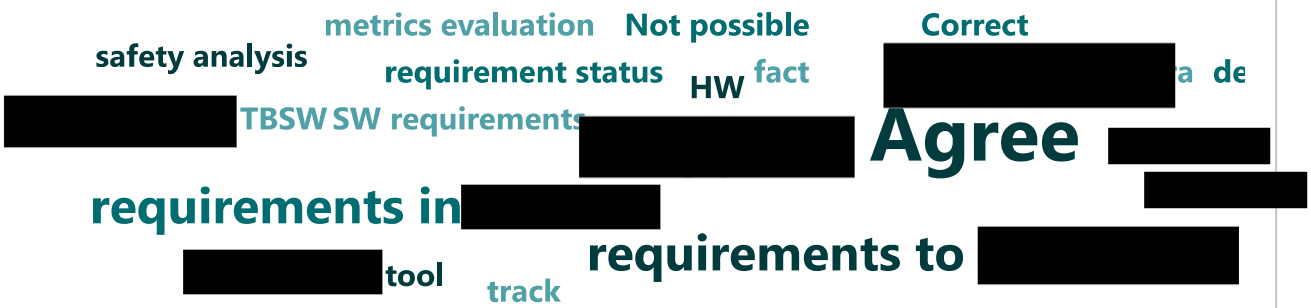
12
Responses

Latest Responses

"Agree"

"Correct! Maybe we don't write the requirements [redacted] but keep..."

3 respondents (**25**%) answered [redacted] for this question.



5. Please state if you disagree with anything that we mentioned about **"traceability tools/ practices"** or if there is something else you want to add?

11
Responses

Latest Responses

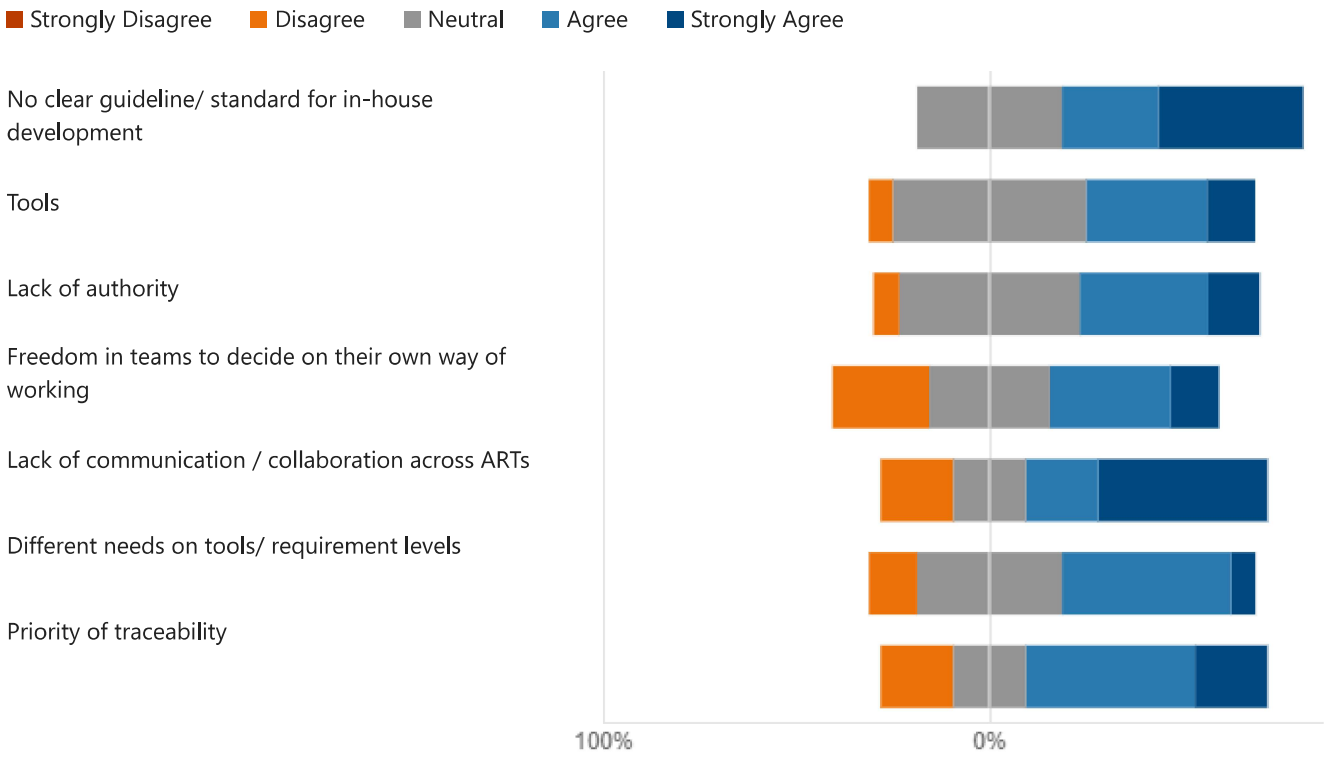
"Agree"

"Nothing to add or comment."

4 respondents (**36**%) answered **Simulink** for this question.

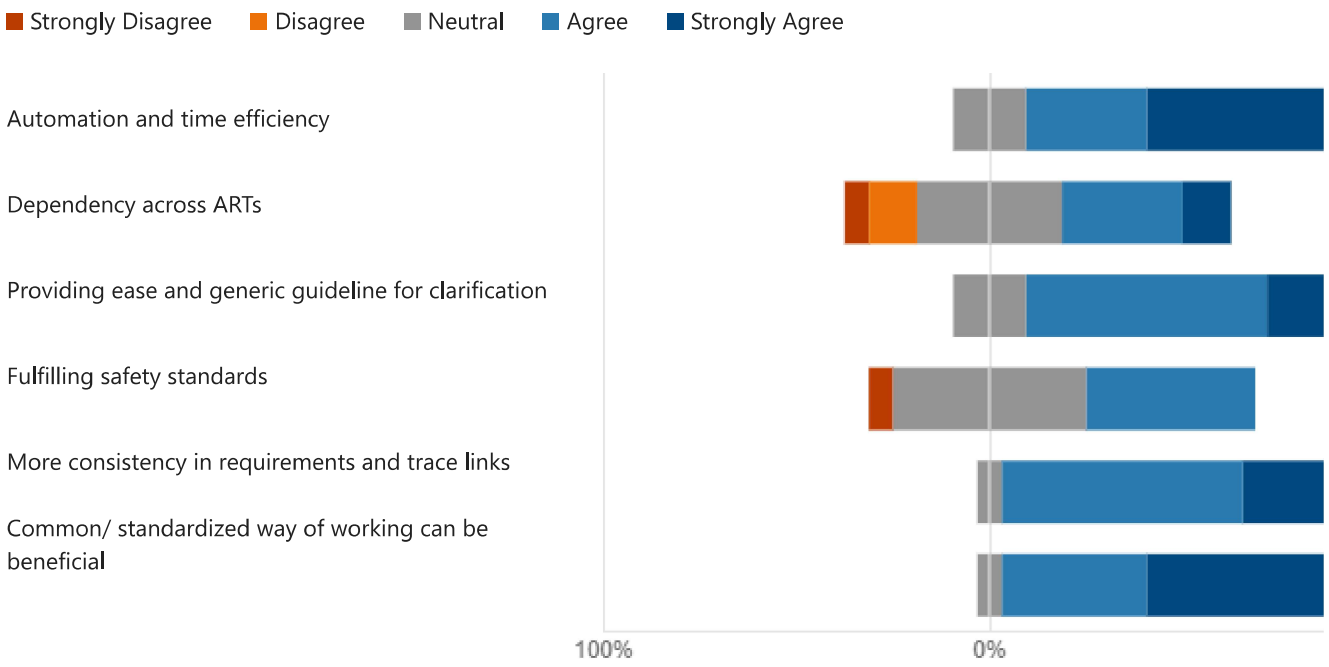
6. Please indicate your agreement with the following statements:

"One of the reasons behind differences in requirements documentation and traceability practices between ARTs is ...":



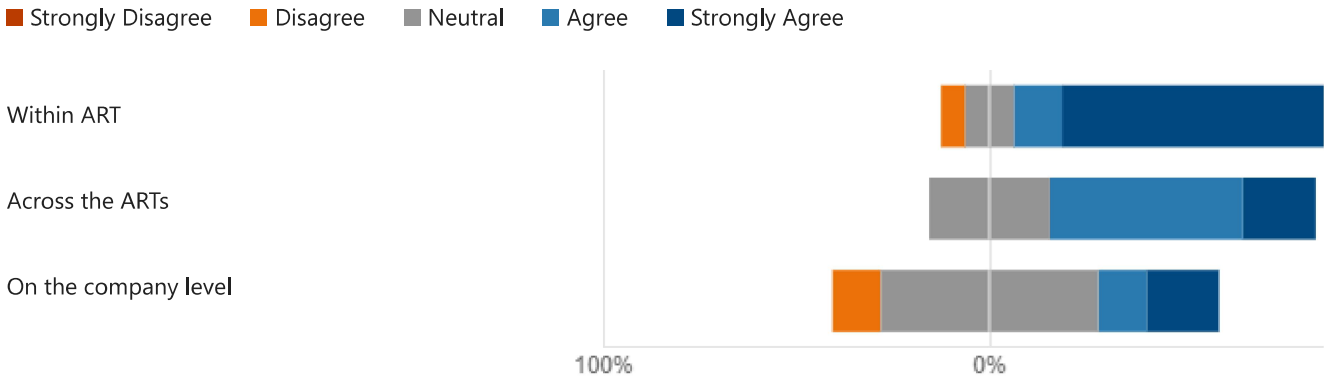
7. Please indicate your agreement with the following statements:

"Moving toward more alignment is required because of ..."



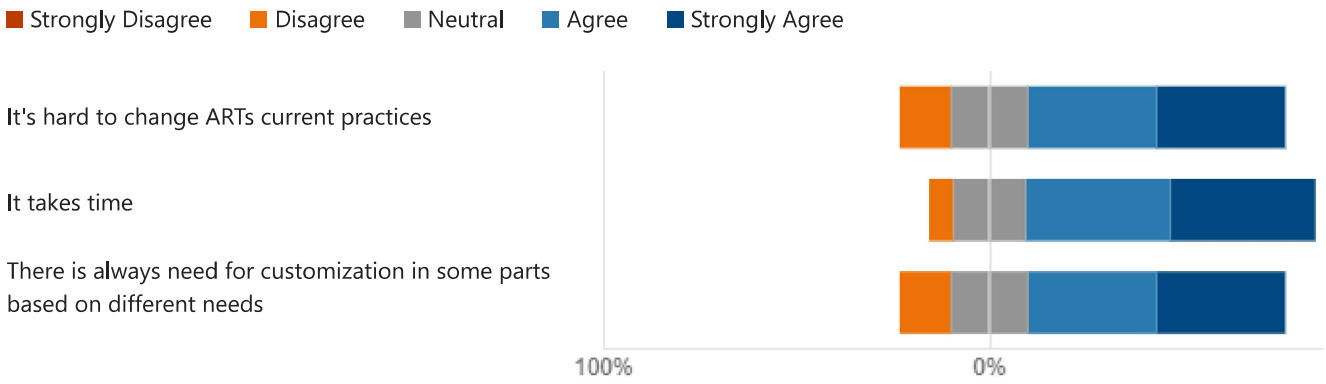
8. Please indicate your agreement with the following statements:

"To help with solving the problems mentioned related to differences of requirements documentation and traceability practices , a more common way of working/ standard needs to be applied ..."



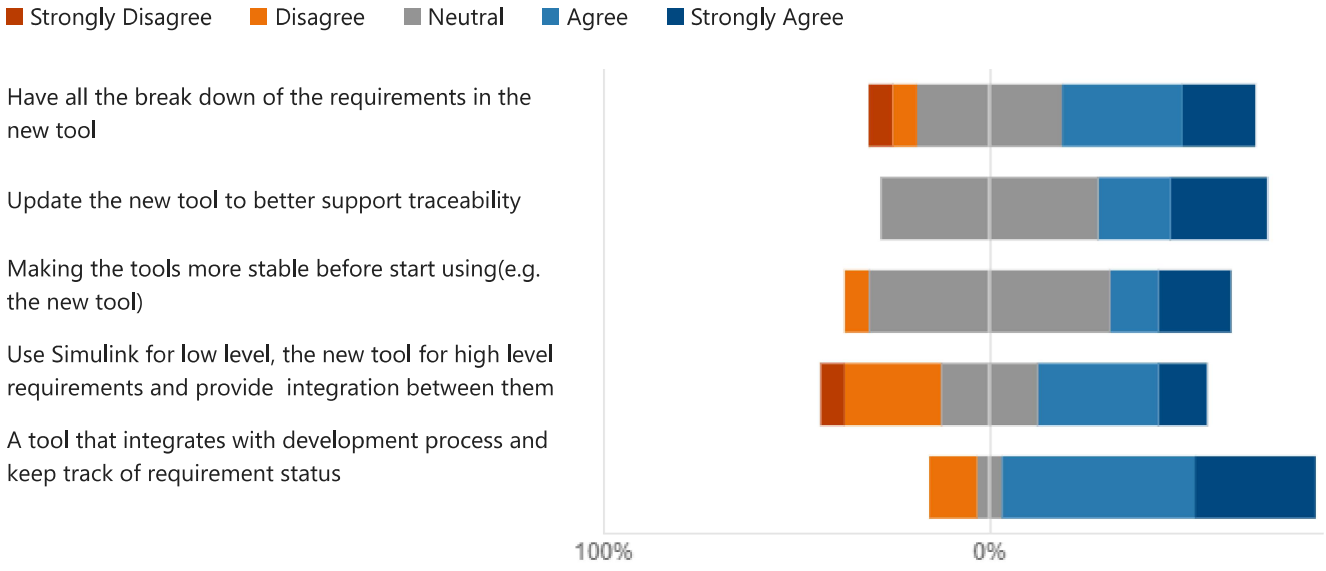
9. Please indicate your agreement with the following statements:

"One of the important **challenges** for moving toward alignment is ..."

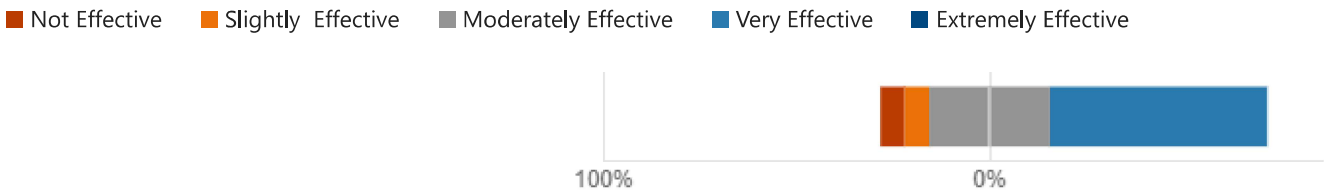


10. Please indicate your agreement with the following statements:

"In order to solve the problems mentioned, related to tools, we need to ..."

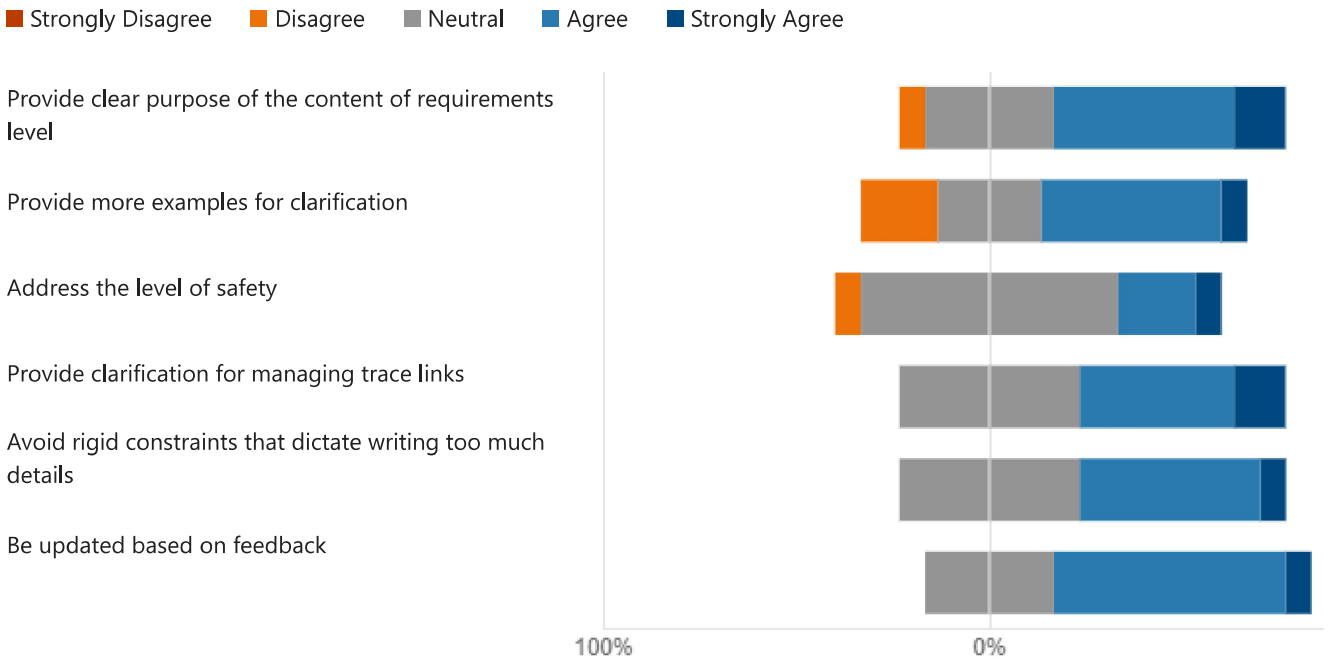


11. How do you think a more mature reference/ standard model for requirements and traceability can be effective in solving problems related to differences in requirement documentation and traceability practices between ARTs?



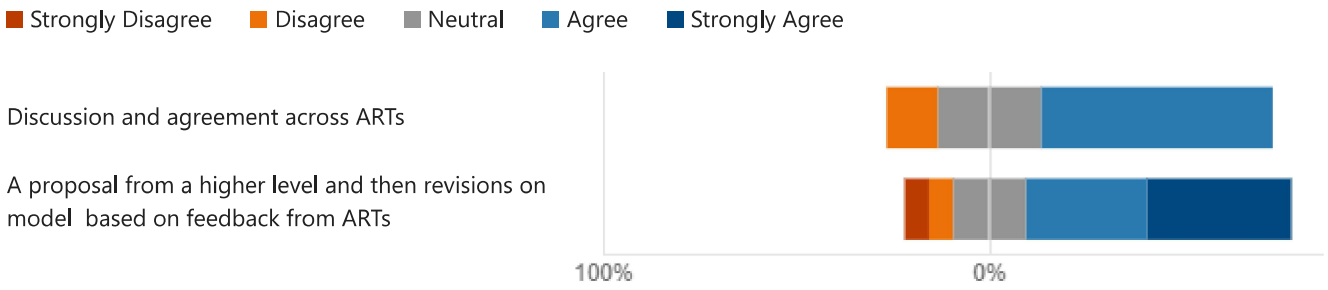
12. Please indicate your agreement with the following statements:

"A more efficient and mature reference/ standard model for requirements and traceability is required to ..." :



13. Please indicate your agreement with the following statements:

"A central initiative for developing a reference/ standard model for requirements and traceability is better to be based on ..."

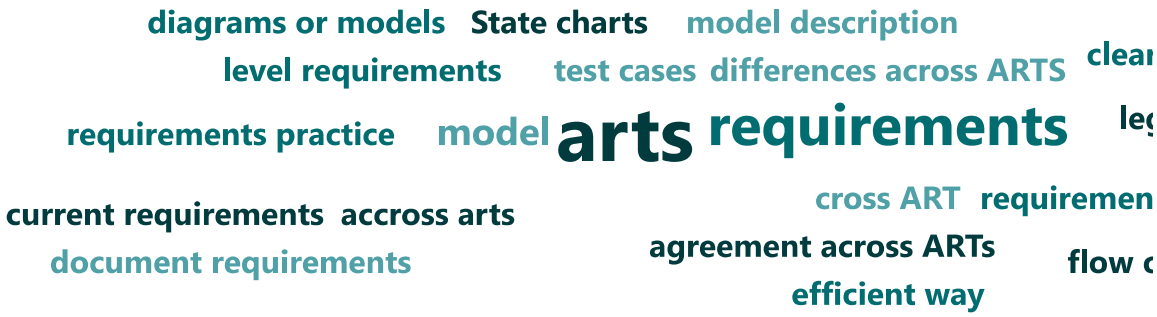


14. Do you have any other suggestion to solve the problems related to differences across ARTs?

6
Responses

Latest Responses
"I'm hesitating in the efficiency of working and agreeing cross A...

4 respondents (**67**%) answered **arts** for this question.



D

Appendix 4

D.1 Coding Process - Step 1

Intervi	Q /	Statement	Category	Code	Comment
1		These are the different levels of requirement.So all the different requirements are captured in the new tool			
1	Q	And do you know about the similarities and differences with other teams in your Art related to this documentation practices?			
1	R	Ah yeah, similarities, I will say, like, it is a generic level,	ARTs' Similarities	generic level of requirements	SWRS
1	R	it is the Company guideline to have the different levels of requirements, so most of them will have all the different levels	Standard	Guideline of different levels of requirements	Company guideline to requirements
1	R	but there can be difference. For exampleART1 may not have a customer function requirement.even ART2 don't have any customer function requirements, only the Art1 will have. The customer function requirement so those kind of things can be vary	ARTs' Differences	different requirement levels between different ARTs	
1	R	but if you go into a bit more lower level of requirement then they need to follow these different levels. So mostly from the PC Or the product function requirement. product capability requirement and they SWrS requirements will be common across Art.			
1	Q	So you you also said the similarities with other arts here and the OK. I can you also tell us about, like the trace links that you are using in your team and the differences with other teams and other ArTs.			
1		Even the trace is kind of in standard way that is defined by the the new tool also how to trace between different levels is also we have a Guideline.	Standard	Traceability Guideline	the new tool
1	R	so in the new tool we have the old way of tracing using the [redacted] Trace	Traceability		the new tool old way)
		but right now for almost like 2 weeks they introduce their new way of tracing it is called as analysis	Traceability	Analysis requirements	

Figure D.1: Spreadsheet of Interviews Codes - Step 1

E

Appendix 5

E.1 Coding Process - Step 2

A	B
Category	Code
ARTs' Differences Reasons	No Collaboration about their needs
ARTs' Differences Reasons	safety
ARTs' Similarities	lower level of requirement(PC, product function, SWRS)
ARTs' Similarities	Company guideline to have the different levels of requirements
ARTs' Similarities	generic level of requirements
ARTs' Similarities	Duplicate Requirements in [redacted] A tool
Change drivers	drawbacks of [redacted] A tool
Change drivers	duplicate requirements (SWRS with SRD)
Change drivers	Same tool for requirement and traceability
Change drivers	to have SRD in higher level
Change drivers	Better Traceability from Implementation to functionality
Change drivers	decouple with the functionality and the logic versus the implementation
Change drivers	decrease manual work
Change drivers	fulfill the safety standards
Change drivers	more common way of working
Change drivers	New tool support the needed data model of the requirements
Change drivers	visiblity of dependency of requiremenet to see change effects
Collaboration with ARTs	Not about requirement practices
Collaboration with ARTs	about requirement practices
Collaboration with supporting ART	about requirement practices(previous)
Collaboration with Supporting ART	not about requirement practices
Customer function	First level of requirement
Customization	ART decide the actual content of level of requirement

Figure E.1: Spreadsheet of Interviews Codes - Step 2

F

Appendix 6

F.1 Coding Process - Step 3

Ro	Interview	Statement	Category	Code	Comment	RQ	Note	Count	Interview Count
500	4	requirement information into a CSV file. And if we just stopped there , there's many things	Traceability	Manual	(human mistake)	1 & 3.1			
1354	14	failures for Latin point or fail or so and for random hardware faults.	Traceability	Manual	analysis to see how if our hardware design together)	1 & 3.1			
1386	14	overtime ,yeah, mainly two issues. Process and tools for low level design.	Traceability	Manual	hard to maintain manual traceability	1 & 3.1			
1436	14	was a very good work done because they try to reduce the effort as much as possible.	Traceability	Manual	hard to maintain manual traceability	1 & 3.1			
1069	8	wait until closer to production started, tried to do a complete summary.	Traceability	Manual	Not efficient because of frequent changes	1 & 3.1			
1336	10	chance that people will will not do it all the time because it's it's too much work.	Traceability	Manual	people don't do it becasue manual	1 & 3.1			
1415	14	something that's done manually over time, it's not followed.	Traceability	Manual	people don't do it when it is manual	1 & 3.1			
1123	8	,usually what happens then is that they don't do it at all.	Traceability	Manual	they do not do that manually	1 & 3.1			
1298	10	on how do you actually get the trace ability without adding manual work?	Traceability	Manual		1 & 3.1			
1356	14	and where we have a manual traceability and other is also for low level design.	Traceability	Manual		1 & 3.1			
1431	14	developed and tested.And that link will need most probably to be done manually	Traceability	Manual		1 & 3.1		11	4

Figure F.1: Spreadsheet of Interviews Codes - Step 3

G

Appendix 5

G.1 Miro Board (The Final step in the coding process in this study)

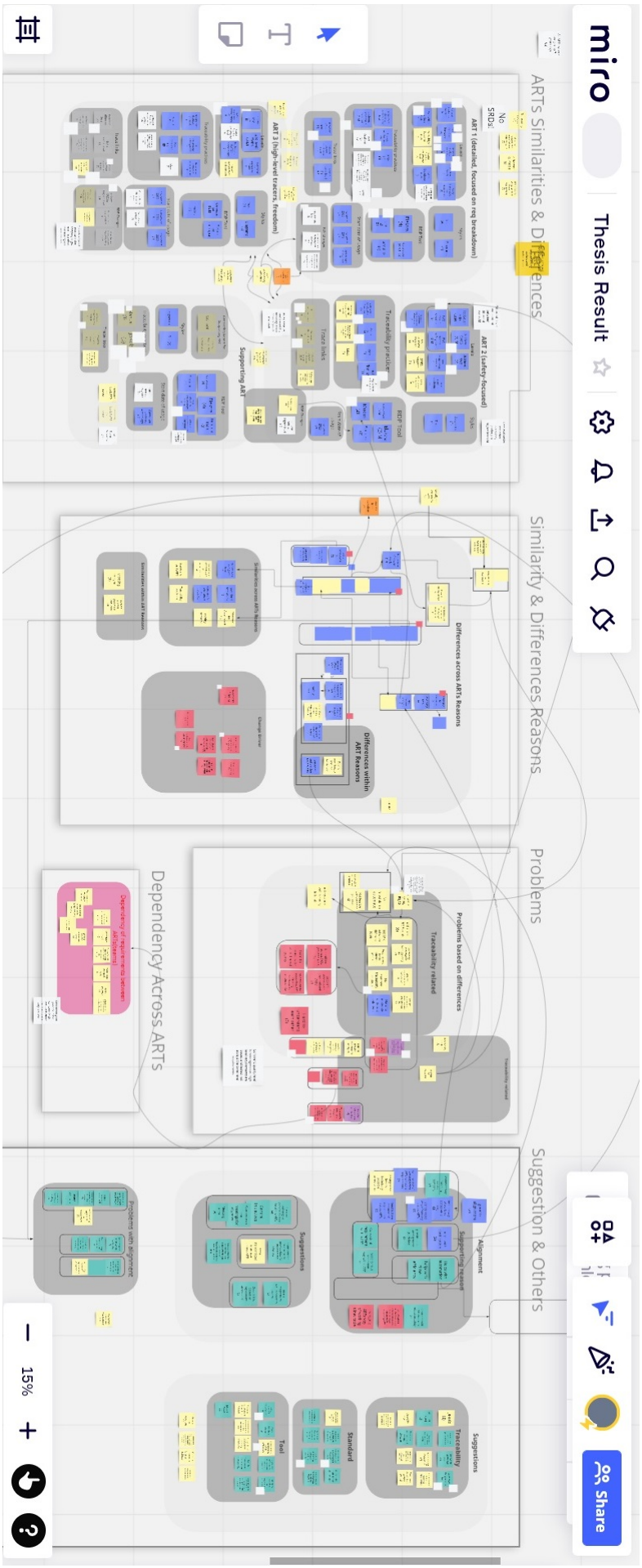


Figure G.1: The study Miro Board for categorizing interviews codes