Journal of Software Engineering Research and Development, 2025, 13:14, doi: 10.5753/jserd.2025.6011

Software Tool Qualification: A Systematic Literature Review

Giovanni César Borges da Costa [Universidade Federal de São Paulo | giovanni.costa@unifesp.br]
Luiz Eduardo Galvão Martins [Universidade Federal de São Paulo | legmartins@unifesp.br]

Abstract

Safety-critical systems are subject to rigorous regulation, demanding compliance with specific criteria before it is made available. In safety-critical domains, this is accomplished through the fulfillment of recognized safety standards requirements or objectives. This SLR has the purpose of identifying approaches or methods to comply with safety standards requirements or objectives related to tool qualification aspects. It also pursued the identification of methods used to demonstrate tool confidence, as well as the activities and artifacts generated to demonstrate that. This study also identified challenges practitioners face to qualify software tools. This SLR was conducted per guidelines proposed by Kitchenham and Biolchini. A search string was developed, and its application returned more than 1,180 papers. For this study, 41 papers were selected, as per the criteria defined in the protocol. These SLR results indicate that there are many studies describing approaches based on established safety standards, for ensuring confidence in tool functionality. The results indicated that research on different proposed methods for tool qualification exists, especially in automotive context. The results also point to the necessity of additional research on toolchain integration. The SLR main findings were the identification of methods for tool qualification, concerns, approaches to address toolchain integration.

Keywords: *Software, Tools, Qualification, Toolchain, Systematic Literature Review.*

1 Introduction

In our day-to-day routines most of us are, at some level, exposed to safety-critical systems. Safety-critical systems are those systems whose failure or malfunctioning may result in harm to the people, harm to the environment or property damage or loss (Knight, 2002; Rushby, 1994).

Due to the grave consequences of its failures, safety-critical systems are subject to strict regulation, demanding compliance with certain acceptance criteria before it is made available for use. Companies and other entities that pursue to deploy safety-critical systems are therefore obliged to demonstrate compliance with some sort of acceptance criteria, which may vary from domain to domain, as established by the regulatory bodies entitled with the task of ensuring safety-critical systems provide acceptable safety levels.

Civil aviation is one of the domains where regulators have established specific acceptance criteria for the certification of airplanes, engines, and propellers. Civil aviation regulators have also identified acceptable means of compliance with the certification requirements, including ones concerning software aspects, which are formalized in regulatory documentation such as EASA AMC 20-115D and FAA AC 20-115D (Federal Aviation Administration, 2017). Both AMC 20-115D and AC 20-115D recognizes DO-178C (RTCA, 2011d) and its supplements - DO-331 (RTCA, 2011a), DO-332 (RTCA, 2011b) and DO-333 (RTCA, 2011c)) - and DO-330 (RTCA, 2011f) as acceptable means of compliance with the applicable certification requirements concerning software aspects of airborne safety-critical systems.

The DO-178C is a RTCA, Inc. standard, which "provides guidance for determining, in a consistent manner and with an acceptable level of confidence, that the software aspects of airborne systems and equipment comply with airworthiness requirements" (RTCA, 2011d). Its supplements DO-331, DO-332 and DO-333 tailor and complements DO-178C objectives to address specific concerns related to model-based development, object-oriented techniques and formal methods, respectively (RTCA, 2011a; RTCA, 2011b; RTCA, 2011c). The document DO-330 addresses the considerations on software tool qualification in the context of safety-critical applications, incorporating the guidance previously provided in DO-178B (RTCA, 1992) document – thus removed in DO-178C – and expanding it to provide a comprehensive, dedicated set of objectives to be fulfilled to ensure tool confidence.

Similarly, acceptance criteria for safety-critical software have been established by the relevant stakeholders for other domains. The United States Food and Drugs Administration (FDA) recognizes the IEC 62304 (Food and Drugs Administration, 2019; International Electronical Commission, 2015b) as a relevant standard for medical device software. For railway applications, IEC 62279 (International Electronical Commission, 2015a) is the standard specifying process and technical requirements for development of software used in railway control and protection applications. In automotive industry, ISO 26262 (International Standard Organization, 2018) is the standard for road vehicles functional safety, which includes specifications for product development at software level.

Software development in multiple domains is usually heavily supported by software tools, assisting various software life cycle processes. Software tools comprehend a wide variety of computer programs that are used through the software life cycle to support the development, verification and control of a program and its documentation. When software tools are used to support life cycle processes of safety-critical software, in such a way that its outputs are granted as correct without further evaluation, it becomes necessary to ensure that introduction of errors or failure to detect errors due to a tool malfunctioning is minimized.

In general, safety-critical software standards applied to different domains address some criteria, requirements, or objectives to ensure that confidence in tool functionality is obtained, so that a user may rely on its outputs to comply with the objectives and criteria established by the applicable standard.

In the context of civil aviation airborne safety-critical software, DO-178C establishes the criteria to determine the required qualification level for software tools, the Tool Qualification Level (TQL), while DO-330 provides the guideline for tool qualification, defining objectives and their related activities to produce evidence that tool functionality can be relied upon. Although the DO-330 was initially conceived for use in the qualification of tools in the aeronautical context, it can be used by other domains and technologies, such as automotive, space, medical, electronic hardware, and aeronautical databases (RTCA, 2011f). ISO 26262 also addresses, in its Part 8, the objectives and guidance on tool qualification to ensure confidence in the functionality of software tools. ISO 26262-8 establishes the criteria to determine the Tool Confidence Level (TCL), based on the impacts a tool malfunctioning may have on a software by either introducing an error to the software or failing to detect one.

This Systematic Literature Review (SLR) was conducted with the purpose of investigating among the existing literature what are the different approaches and methods applied for demonstrating tool confidence, as well as activities conducted, and artifacts generated as part of the tool qualification effort. This SLR also aimed to identify difficulties and challenges encountered to achieve compliance with the safety-standards requirements related to tool qualification aspects. From the data extracted, an analysis and discussion of the SLR results was performed to answer the proposed research questions established for this SLR.

This paper describes the methodology followed to conduct the SLR and records the analysis and discussion of the results found. It is organized as follows: Section 2 establishes some background definition and discusses about related works. Section 3 describes the methodology applied for this SLR. Section 4 presents the analysis and discussion of the results on answering the proposed research questions. Section 5 presents the conclusions from this SLR.

2 Background and Related Work

2.1 Definitions

The Introduction section of this paper describes the reasoning for establishing acceptance criteria that must be accomplished, as enforced by regulations, to allow the deployment of safety- critical systems for public use, as well identifies a few standards recognized as acceptance criteria for approving safety-critical software. A variety of these standards provide guidelines on acceptance criteria when software tools are utilized to automate, reduce, or even eliminate development or verification activities. Since the main subject of this SLR is the identification of different approaches and methods applied for demonstrating tool confidence, as well as activities conducted, and artifacts generated as part of the tool qualification effort, the following terms and definitions should be considered:

Safety-Critical Systems: Safety-critical systems are those systems whose failure or malfunctioning may result in harm to the public such as injury or death to humans, significant damage to property or to the environment (Knight, 2002; Rushby, 1994). Due to the grave consequences that may result from safety-critical systems failures, these systems must be designed to address safety objectives so that it functions in safe manner. That implies in design that encompasses architectural arrangements and strategies that can both limit the adverse impacts due to failures and detect failures and respond safely to them (RTCA, 2011d; SAE International, 2012). Having in place adequate and comprehensive safety and requirements engineering processes are crucial factors for the successfully design of safety-critical systems. These demands on the safety- critical systems development are enforced by the strict regulation these systems are subject to.

Tool qualification: According to the definition of DO-330, tool qualification is "the process necessary to obtain certification credit for a software tool within the context of a specific system". According to ISO 26262, tool qualification is the process "to create evidence that the software tool is suitable to be used to support the activities or tasks required by the ISO 26262 series of standards (i.e. the user can rely on the correct functioning of a software tool for those activities or tasks required by the ISO 26262 series of standards)."

Practitioner: A tool developer or user responsible to conduct the activities and generate the artifacts needed for demonstrating compliance with the safety-standards requirements related to tool qualification aspects. It may also be a compliance verification specialist, in charge to verify that the compliance with the safety-standards requirements related to tool qualification aspects have been properly demonstrated.

Approach: An approach is a conceptual, higher level and abstract way to rationalize the stream of actions to resolve a given problem or situation.

Method: A method is a specific set of actions, concrete and specific procedures to address a given problem.

2.1 Related Work

Prior to the deployment of a safety-critical system, assurance must be obtained that compliance with the applicable acceptance criteria established by regulatory bodies is achieved and properly demonstrated. It encompasses the compliance with safety standards requirements or objectives applicable for software embedded into safety-critical systems, including those related to tool qualification.

In the scope of this SLR, it was not identified studies addressing a systematic review of approaches or methods used to demonstrate compliance with safety-critical standards requirements or objectives related to tool qualification aspects. A few related works identified in the literature, discuss approaches, methods and challenges to demonstrate compliance with tool qualification related requirements or objectives.

Gallina *et al*. (2014) study presents an approach to allow the reuse of tool qualification artifacts from one domain to another. The study describes that multiple safety standards address the concerns related with the use of software tools that may introduce errors or fail to detect them. This concern may be addressed through the qualification of the software tools, for which processes and criteria are defined by safety standards. The paper presents an approach to achieving tool qualification, through the reuse of tool qualification artifacts for other domains. This is based on the establishment of a tool qualification process line and the use of process-based argument, using GSN (Goal Structuring Notation) for documenting safety cases.

Marques and Cunha (2017) study provide a description of the evolution of the tool qualification subject in the aviation domain. According to that study, tool qualification has become a more critical issue of concern when the safety standard introduced objectives that could only be feasible with the use of software tools.

The SLR presented in this paper discusses the activities conducted and artifacts produced for qualification of a software tool, identifying some challenges and pitfalls faced by practitioners to properly achieve and demonstrate compliance with tool qualification requirements.

3 Research Methodology

This Systematic Literature Review was performed considering the research methodology guidelines described in Biolchini *et al.* (2005), Kitchenham & Charters (2007), and Martins & Gorschek (2016). A description of the methods and criteria adopted to conduct the SLR is presented in the upcoming sections – Figure 1 depicts the methodology followed to conduct the SLR.

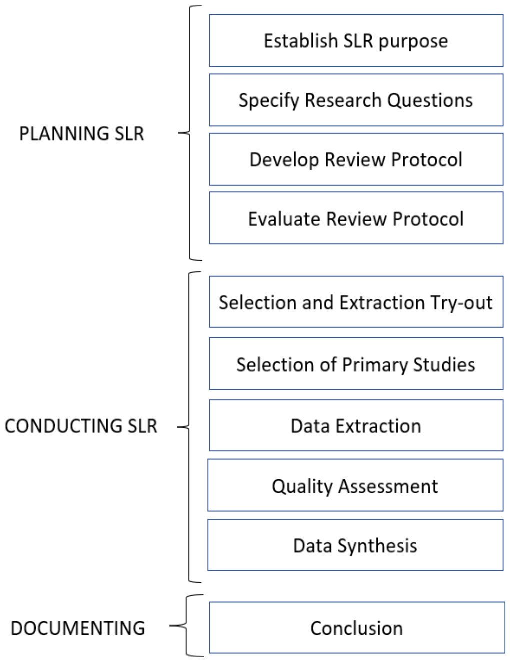

Figure 1. Systematic Literature Review steps adapted from [42] [51]

The purpose of this systematic literature review was to the identify different approaches and methods to ensure confidence in tool functionality, as well as identifying the activities conducted and artifacts produced as part of the tool qualification effort. This SLR also aimed to identify difficulties and challenges encountered to demonstrate compliance with the safety standards requirements related to tool qualification aspects. From the data extracted, an analysis and discussion of the SLR results was performed in order to answer the proposed research questions established for this SLR.

Aiming to verify whether other similar literature review have been previously conducted or not, a search was conducted in the digital libraries selected as source of data for this SLR:

- ACM Digital Library
- IEEE Xplore
- Science Direct
- Springer Link.

Studies titles, abstracts and keywords were scanned using the following search string:

("Tool Confidence") OR ("Tool Qualification") AND (Review OR SLR OR Research overview)

The resulting studies selection obtained from this query, although related with the subject of software tool qualification, did not provide for answering all the research questions established as part of the review protocol. The research questions determined for this SLR, presented in Table 1, are aimed to direct the data extraction from the selected studies and the discussion and synthesis to be conducted for concluding the SLR.

Table 1. SLR Research Questions

ID	Research Question	Purpose
RQ1	What are the approaches considered to demonstrate tool confidence in the context of safety-critical software approval?	To identify the main approaches considered to ensure tool confidence.
RQ2	What are the methods applied to ensure tool confidence in the context of safety-critical software approval?	To identify methods described in the studies selected that are used to demonstrate tool confidence.
RQ3	What are the activities performed to demonstrate tool confidence in the context of safety-critical software approval?	To identify detailed activities performed to demonstrate compliance with applicable tool qualification requirements.

RQ4	What are the artifacts generated as evidence of compliance with tool qualification requirements?	To identify artifacts generated as evidence to demonstrate compliance with applicable tool qualification requirements from safety standards.
RQ5	What are the main challenges identified to demonstrate the proper achievement of tool qualification requirements?	To identify difficulties and challenges to demonstrate compliance with tool qualification requirements.

3.1 Search Strategy

The strategy for the identification of the papers for this SLR followed the steps depicted in Fig. 2. This strategy is based on the guidelines for SLR proposed by Kitchenham & Charters (2007), and Martins & Gorschek (2016).

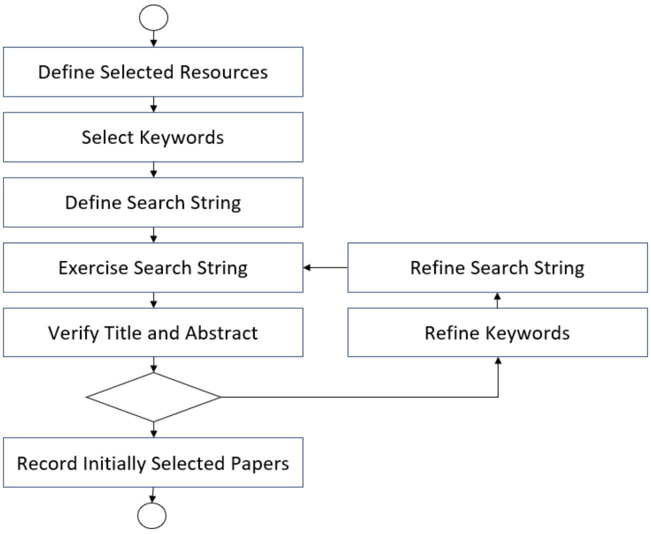

Fig. 2. SLR Search Strategy adapted from [51][53]

The definition of the search string was primarily based on the research questions. The combinations of words in the initial search string aimed to optimize the search from the databases, looking for studies that were related with safety-critical software, addressing tools and tools qualification, in the context of approval or certification. The following string was initially proposed:

((safety-critical software) OR (critical software) OR (safety software)) AND ((software tool) OR (tool)) AND ((tool confidence) OR (tool qualification)) AND ((approval) OR (certification))

This initial string was exercised on IEEE Xplore and Science Direct databases, to verify the adequacy and efficacy of the string proposed, in terms of returning relevant studies for the SLR. It was observed at first that the inclusion of the terms "approval" and "certification", aimed to narrow the results to those related with software approval or certification, was ineffective – there were no significant changes on the search results when adding or removing these terms.

A variety of different terms denoting "safety-critical" software and system were used in the search strings exercised, aiming to ensure that a broader range of related publications in the databases is identified. The exercise showed that the terms "safety-critical software" and "safety-critical system" did not contribute to identify additional publications than the ones identified with the search string using the terms "critical software" and "safety software". In equivalent manner, the search string exercise identified that the term "tool" is covered by the term "software tool".

Finally, the search string was also exercised with terms that could be considered as synonyms to "qualification", such as assurance, validation, accreditation. The exercise demonstrated that the term "accreditation" does not affect the search results, thus it was not added to the search string. Additional try-outs showed that the search results are more precise when key terms are quoted. The following search string was defined to be exercised in each database:

((critical software) OR (safety software)) AND ((software tool)) AND (("tool confidence") OR ("tool qualification") OR ("tool assurance") OR ("tool validation"))

Aiming to focus the search on each database, the search string was exercised and customized based on the results found. It was then conducted the search in the following databases, considering the fields as described:

ACM Digital Library:

- In "Advanced Search", selected "Search items from: The ACM Guide to Computing Literature"
- In "Search Within" was selected "Anywhere"
- Used the following string, search expression:

((safety-critical software) OR (critical software) OR (safety software) OR (safety-critical system)) AND ((software tool) OR (tool)) AND (("tool confidence") OR ("tool qualification") OR ("tool accreditation") OR ("tool assurance") OR ("tool validation"))

IEEE Xplore:

- In "Advanced Search", selected "Command Search".
- In "Data Fields" was selected "Full Text & Metadata".
- Used the following string, search expression:

((("Full Text & Metadata":safety-critical software) OR ("Full Text & Metadata":critical software) OR ("Full Text & Metadata":safety software) OR ("Full Text & Metadata":safety-critical system)) AND (("Full Text & Metadata":software tool) OR ("Full Text & Metadata":tool)) AND (("Full Text & Metadata":"tool confidence") OR ("Full Text & Metadata":"tool qualification") OR ("Full Text & Metadata":"tool accreditation") OR ("Full Text & Metadata":"tool assurance") OR ("Full Text & Metadata":"tool validation")))

- Filters Applied: Conferences and Journals.

Science Direct:

- In “Advanced Search”, field "Find articles with these terms”.
- Used the following string, search expression:

((critical software) OR (safety-critical system)) AND ((software tool) OR (tool)) AND (("tool confidence") OR ("tool qualification") OR ("tool accreditation") OR ("tool assurance") OR ("tool validation"))

- Filter Applied: Subject Areas “Engineering” and “Computer Science”.

Springer Link:

- In “Advanced Search”, selected “Find Resources: with all of the words”.
- Used the following string, search expression:

((safety-critical software) OR (critical software) OR (safety software)) AND ((software tool) OR (tool)) AND (("tool confidence") OR ("tool qualification"))

- Applied filter in "Refine your search" - "Content Type": Conference Paper.
- Applied filter in "Refine your search" - "Content Type": Article, Chapter.

3.2 Review Protocol

Following the research methodology as depicted in Figure 1, a review protocol was established. First step, the resources from where the studies would be selected were defined. The digital libraries selected as source of data for this SLR, as already mentioned were ACM Digital Library, IEEE Xplore, Science Direct and Springer Link.

The search engines available in the websites of these digital libraries was used to find the applicable papers, utilizing the search string defined for this SLR. Additional refinement on the search was done using other filters and features provided in these digital libraries search engines.

For this SLR, the **population** considered were only publications addressing the subject of qualification of software tools utilized in the context of safety-critical applications, while the objective of the **intervention** was to identify approaches and methods used to demonstrate tool confidence, as well as activities performed, and artifacts generated to demonstrate tool confidence the challenges associated with this task.

Table 2. SLR Inclusion and Exclusion Criteria.

	Inclusion Criteria
1	Studies that discuss, describe, or propose approaches or methods to ensure tool confidence.
2	Studies that identify activities performed to demonstrate tool confidence.
3	Studies that describe evidence produced to demonstrate tool confidence.
4	Studies that discuss challenges and difficulties faced by tool qualification stakeholders.

	Exclusion Criteria
1	Duplicated studies
2	Studies not focused on approaches, methods, or activities for tool qualification

Table 2 identifies the inclusion and exclusion criteria established for the selection of the studies. While exercising the search string and the criteria for inclusion or exclusion of papers, other refinements on the search were also considered. In IEEE Xplore library, only Journals and Conferences articles were considered. In Science Direct, only research articles from publications related with computer science and safety-critical applications were considered.

3.3 Procedure for Studies Selection

Once the protocol and criteria for the selection of the studies from the digital libraries was established, the procedure for the actual selection of the applicable studies was applied. Figure 3 depicts the procedure followed for this activity.

The search string defined for this SLR was applied in each of the databases selected, with the fine-tuning filtering as described in review protocol. The outcome of this search was then subject to the assessment of the inclusion and exclusion criteria. Each of the studies returned had its Title and Abstract read and verified against the Exclusion criteria – the study is excluded in case the title and abstract are referred to an study that does not address tool qualification approaches (i.e. does not describe a high level concept on how to obtain confidence in the tool’s functionality), tool qualification methods or activities (i.e. does not describe a detailed methodology or the activities to obtain confidence in the tool’s functionality), or the study is already added to the list of primary studies.

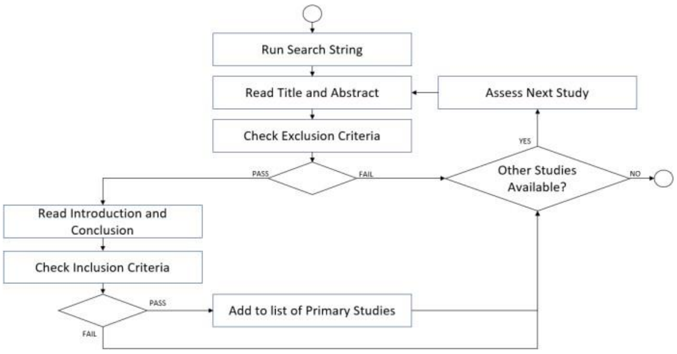

Figure 3. SLR Procedure for selecting studies.

In case the study passes this first assessment, or in those cases where this assessment was inconclusive, the paper’s Introduction and Conclusion was also read to verify the relevance of the study for the SLR. In case the papers address the Inclusion criteria, it was added to the list of primary studies for this SLR. This process continues while there are still studies available, until the end of studies list returned from the search string. A total of 1,180 studies returned from the initial search applying the search string and filtering in the applicable databases and were assessed as per the procedure described above. The first assessment of the inclusion and exclusion criteria resulted in the selection of 111 studies. The second assessment, which comprises of a more detailed review of the papers, at minimum its introduction and conclusion, was conducted and resulted in the list of papers that are more relevant to this SLR. Table 3 provides the information on the distribution and the total of studies considered relevant for this SLR.

Table 3. SLR Studies Selection Result.

Database	Initial search	1st Round Inclusion criteria	2nd Round Inclusion criteria
ACM Digital	155	11	6
IEEE Xplore	305	53	22
Science Direct	284	13	6
Springer Link	436	34	7

3.4 Data Extraction

As mentioned before, the application to the defined search string in the selected digital libraries databases returned a total of 1180 studies, distributed as presented in Table 3. Following the procedure for studies selection and applying the inclusion/exclusion criteria, a total of 111 studies were initially selected and then a more detailed assessment resulted in the selection of 41 studies. These 41 studies were selected for data extraction and more comprehensive review in order to answer the research questions aimed for this SLR.

As part of the protocol for the studies selection while conducting the SLR, a pre-determined list of data of interest to be acquired from the papers selected was defined, which is presented in Table 4. Throughout the assessment of the studies selected, these data are extracted from the papers and provide valuable information to answer the Research Questions (RQ). For that reason, Table 4 identifies the RQs related with each data, when applicable.

Table 4. SLR Data Extraction.

Data	Description	Related RQ
Study Identifier	Means to uniquely identify paper	None. Paper overview only.
Library	Source of paper	None. Paper overview only.
Title, Year of Publication, Author	Basic information on paper	None. Paper overview only.
Research Method	Research method applied in the paper. It may be, for instance, Case Study, Conceptual Analysis, Experiment, Industry Report, SLR, Survey.	None. Paper overview only.
Types of tools	Identifies the types of tools	None.
Domain	Paper study domain. It may be, for instance, Aerospace, Automotive, Industrial, Medicine, Safety, Railway, Digital Forensic, System Engineering	None. Paper overview only.
Software Tools	addressed in the studies – it may be either a tool that may introduce an error (for short, named "Development"), a tool that may fail to detect an error (for short, named "Verification"), or both.	Paper overview only.
Approaches or Methods used for Tool Qualification	Identifies methods or approaches applied for tool qualification. For instance, applying a process-based approach, such as DO-330; or ensuring tool confidence through Assurance Cases.	RQ1 RQ2
Safety-Critical Standard	Safety standard considered in the paper, for instance: DO-178B, DO-178C, DO-200B, DO-278A, IEC 62279, IEC 62304, ISO 26262, DO-330, ISO 17025, ISO 61508	RQ1 RQ2
Activities conducted / Artifacts produced to qualify tool	Identify activities conducted to qualify tool, such as tests, analysis, etc., as well as artifacts produced to qualify tool, such as models, requirements, test cases, test results, assurance reports, etc.	RQ3 RQ4
Challenges in tool qualification	Identify main challenges in tool qualification.	RQ5

3.5 Study Quality Assessment

A key step during the execution of the SLR is the assessment of the quality of the studies selected. Answering a set of quality assessment questions, for which are attributed weights related to how completely a study addresses some aspect, it is possible to identify those studies that provides findings or conclusions that are more relevant to substantiate the results of this SLR. The study quality assessment was performed by answering the questions presented in Table 5. These questions were based on the guidelines for SLR proposed by Kitchenham & Charters [51], and Martins & Gorschek [53].

In this SLR, the quality assessment questions were answered for each of the studies selected. As per the guidelines for SLR proposed by Kitchenham & Charters [51], these questions were established to support assessing the quality of the data extracted. These questions could be answered "Yes" when the question was directly and completely answered by the study. The questions were answered "Partial" when the quality aspect assessed is only addressed in some part. It should be answered "No" when the study does not provide answer to the question at all.

Table 5. Study Quality Assessment Results.

ID	Question	Yes	Partial	No
QA 1	Is the study goals clearly explained?	18 (63.4%)	12 (29.6%)	3 (7%)
QA 2	Is the study providing answers to SLR research questions?	3 (7.7%)	25 (68.3%)	10 (24%)
QA 3	Is it clear in which context/domain the study was carried out?	37 (90.3%)	0	4 (9.7%)
QA 4	Are the approaches / methods applied for tool qualification clearly described?	22 (53.8%)	5 (12.2%)	14 (34%)
QA 5	Are pros and cons identified for the approaches/methods used?	3 (7.3%)	0	37 (92.7%)

From the quality assessment performed, it can be observed that there is a lack of studies addressing the research questions in its completion. Although a sizable number of the studies describe possible methods and approaches to demonstrate tool confidence, very few studies describe pros and cons of these proposed methods or approaches. A small number of studies describe activities performed or artifacts produced, as well as challenges faced by practitioners to achieve tool qualification

3.6 Threats to Validity

A possible threat to the validity of this SLR could be resulting from publication bias. As explained by Kitchenham & Charters (2007), studies describing "positive results are more likely to be published than negative results". That possible threat has not seemed as adversely impacting this SLR, since the research questions are not focused on performance results. Moreover, the concept of positive or negative is also a matter of perspective for one evaluating the studies.

Another mitigation factor for this publication bias, was the selection of the resources from where the studies were acquired. The four digital library databases selected for this SLR provide a diversity of vehicles, publishers and reviewers that enhances the confidence on the studies selected.

Another aspect that could be a threat to the validity of the SLR was related to the very specific subject addressed in this literature review. There are several elements of concern when developing, verifying, approving and maintaining safety- critical systems, the topic "Software Tool Qualification" being one of them, a specific concern on software approval in the context of safety-critical systems approval or certification. Thereby, it is plausible to consider that there could be not many studies addressing the aspects captured in the research questions.

4 Results and Analysis

For this SLR, studies discussing distinct aspects of tool qualification were analyzed. These studies applied a variety of different research methods, as case studies, industrial reports, report, conceptual analysis, and survey.

Most of the studies assessed discusses approaches or methods for ensuring confidence in tool functionality or provides some overview on established guidelines for obtaining the confidence that tool functions as intended, composing a total of 13 of the 41 studies assessed.

Another eight studies discussed methods, concerns, and challenges towards the qualification of tools applied in Model-based design activities, either at software or system levels. The application of Formal Methods in tool qualification activities, as well as the qualification of Formal Methods tools, was another topic frequently addressed, being the object of discussion of five of the studies assessed. The discussion around how to deal with the management and qualification of toolchain was also a subject of interest, being addressed in six of the studies considered in this SLR. These topics - qualification of model- based design, formal methods and toolchain tools appear to be a matter of relevance for study on the tool qualification subject. The prevalence of discussion around these three topics reflects the increased use of continuously bigger and complex set of tools integrated that compose the modern environments for software development and verification, as well as the advance and broader adoption of model-based design techniques and formal methods in the field of safety-critical applications. Figure 4 presents the distribution of the main topics discussed in the studies analyzed.

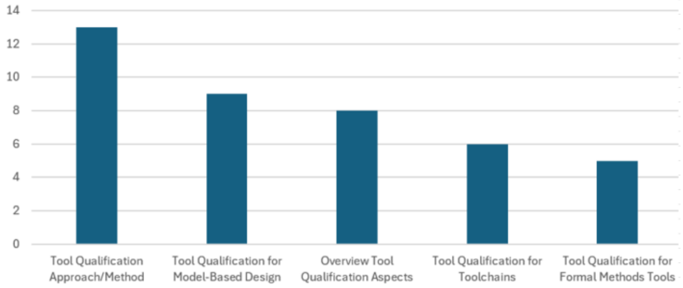

Figure 4. Main topics discussed in selecting studies.

As mentioned, a significant number of studies analyzed in this SLR (21 of a total of 41 studies) either provided an overview on tool qualification aspects or discussed tool qualification methods used, ranging from process-based approach addressing objectives and requirements from established standards such as DO-178C/DO-330 or ISO 26262 to more alternative ways such as the ones proposed for digital forensics tool validation efforts.

Figure 5 depicts some of the approaches or methods discussed in the studies selected – note that these methods were discussed not only in the papers identified as having its focus of discussion on Tool Qualification Methods.

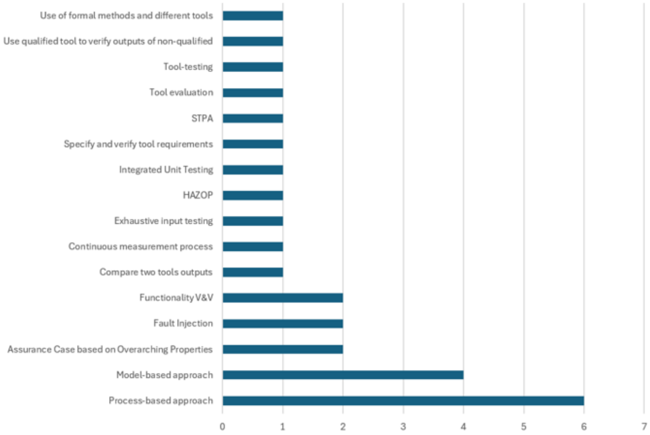

Figure 5. Tool Qualification Approaches and Methods discussed in selecting studies.

The prevalence of studies (Figure 6) related to the Aerospace domain is noticeable and reflects the fact this domain has a more established and detailed guideline for ensuring confidence in tool functionality, provided in DO-254, DO-178B, DO-178C and DO-330 standards. Tool Qualification topic is also of interest of the Automotive domain, as it can be illustrated by the eight studies from this domain.

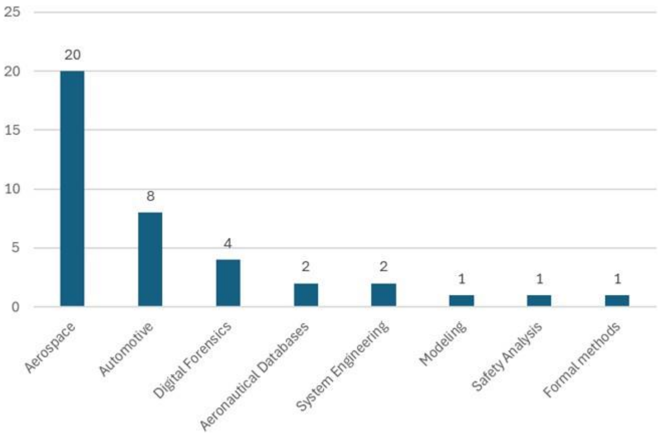

Figure 6. Domains addressed in selected studies.

4.1 RQ1 and RQ2 – Approaches or Methods Considered to Demonstrate Tool Confidence

The purpose of these two research questions was to identify the approaches and/or methods described in the studies selected to ensure tool functionality, thus demonstrating the required tool confidence.

Different approaches or methods proposed to demonstrate tool confidence were identified among the studies selected for this systematic literature review. Part of the studies selected described process-based approaches based on the guidelines of DO-178C and DO-330 or ISO 26262 to ensure confidence in the tool's functionality (Mishra, 2013; Marques e Cunha, 2015a; Marques e Cunha, 2015b; Taft *et al.,* 2017; Schuster, 2022; Tietz *et al.,* 2022; Kornecki and Zalewski, 2008). According to Figure 5, there are several other studies that discusses approaches or methods for ensuring tool confidence, either as its main subject or briefly describing it, proposes or discusses some form of alternative methods to demonstrate tool confidence. These alternative methods to achieve tool qualification may be grouped in three main categories of approaches proposed: model-based; systemic; fault injection. Brauer et al. (2012) proposes a model-driven approach and discusses a method for qualification of a model-based test tool. Aiming to identify and propose a combination of different verification activities that can produce outputs that supports the claiming that the model-based test tool study was reliable, the study describes a strategy for the tool qualification. Firstly, it was analyzed the properties of model-based test tools that relates with requirements for tool qualification. The applicable tool qualification activities are defined in accordance with the level determined for qualification, which is determined based on the criteria specified in the applicable regulation. In the case of ISO 26262, it depends on the Tool Impact and Tool Error Detection Capability - a combination of these two aspects determines the integrity level required. This qualification method proposed for this model-based test tool, although adequate and acceptable to address ISO 26262 tool confidence requirements, cannot be employed in the same manner for DO- 178C compliance, since DO-330 requires a more structured, comprehensive planning and verification of all tool capabilities, not only those more critical functionalities as proposed in that study. Asplund *et al.* (2012) study also describes a model-driven approach and presents a methodology for automatizing the qualification of a toolchain. The study describes a proposed toolchain qualification approach encompassing four steps, from which safety goals are identified. The study argues that the application of the methodology proposed provides for the identification of the toolchain's parts that are more relevant for tool qualification, as well as identifying any extra effort needed to address the integration of toolchains. The study addresses specifically the second step of the methodology, concerning the identification of the required safety goals.

The proposed model-driven approaches described in the studies of Brauer *et al.* (2012) and Asplund *et al.* (2012) rely on modeling techniques to identify and exercise parts of the tools that are deemed more relevant for tool qualification. Some other studies described a more systemic approach to demonstrate the outputs of a tool or a chain of tools, are reliable so that certification credit can be taken.

Baumgart *et al.* (2021) study describes a qualification approach for a chain of tools. In that study, they claim that the existing guidance for tool qualification in different safety standards, such as ISO 26262 and DO-178C, are applicable for single tools, but are not adequate for a toolchain. Baumgart et al. argues that as the list of tools used in a software development becomes bigger and integrates tools from vendors as well as those developed in-house, connected in a complex and automated chain, associated with the lack of guidance from the standards for toolchains, some automation is necessary for the qualification process to ensure this aspect is properly addressed. In one of the case studies described in their work, some characteristics of the toolchain were identified, such as variability (toolchain support a variety of products), efforts to reduce user errors (supported by the use of scripts to glue tools together without the need of human intervention when applying the tools) and evolution (adjusts necessary to support new products).

The study describes the methodology proposed to manage and document a software toolchain. Once the toolchain landscape is known, the next step proposed is to analyze the toolchain to identify risks related. The study proposes the use of the Hazard and Operability Studies (HAZOP) as a methodology to identify the more critical tools in the chain. HAZOP methodology was utilized to determine which tools are critical and to identify possible mitigations. The study highlights that manually managing a complex toolchain and keeping tracking of its configuration is time consuming and error prone.

Asplund *et al.* (2012) study also discusses a system approach to address the qualification of toolchains. Similarly to Baumgart et al. [5] study, it describes that the existing guidelines for tool qualification presented in multiple safety standards, like DO- 178C and ISO 26262, are focused on the qualification of tools isolated, mostly ignoring the safety concerns introduced when these tools are integrated in a so call toolchain. The study argues that the existing tool qualification guidelines presented in current safety standards were appropriate when the development environments were less integrated, with any possible integration being manually carried out and managed by the tools' users. However, this is not the case for modern development environments, where not only an increasing number of tools are employed to support development tasks, but also these tools are automatically integrated. The study proposes the adoption of a system approach, the qualification effort shall encompass the complete toolchain, rather than the collection of tools that compose the chain. The study describes that a System-Theoretic Process Analysis (STPA) was performed on a set of toolchains used in a case study. The study describes the STPA steps and obtained a set of Safety Goals for the toolchain. The study concludes by describing a methodology to identify what need to be qualified as per tool qualification guidelines and what would require additional effort to address toolchain integration concerns.

The study conducted by Gleirscher *et al.* (2023) aimed to identify the main challenges faced to get formal methods tools qualified in accordance with the safety-critical standards. The study focuses on proof assistants, model checker and code generators formal development and verification tools. The study applies the SWOT approach to identify Strengths (what tool qualification benefits tool qualification brings), Weakness (difficulties and barriers for achieving qualification), Opportunities (what are the foreseen possibilities of improvement when pursuing tool qualification) and Treats (what could make tool qualification harder to achieve). From this analysis, the authors draw their suggested approach to easy the path towards achieving tool qualification.

Waldvogel (2022) study discusses a proposed approach to qualify a domain-specific modelling environment including graphical editor. The study identifies that the main challenge in qualifying a domain-specific modelling tool is its complexity, which would demand bigger effort. The proposed approach aims to facilitate qualification by assigning different tool qualification levels to distinct parts of the tool. The study describes that a higher TQL could be assigned for the portion of the modelling tool that could introduce some error. Less stringent TQL may be assigned to visualization portion of the modelling tool. The proposed approach includes that the interface among the different tool portions shall be qualified accordingly, to ensure that less stringent qualified portions do not adversely impacts into the more stringent qualified portions.

As depicted in Figure 4, the use of toolchains and the associated concerns and challenges it imposes to tool qualification efforts is an important topic of discussion in the literature. As recognized by Asplund *et al.* (2012) and Baumgart *et al.* (2021), the existing tool qualification safety standards, such as DO-330 and ISO 26262, do not provide sufficient guidance on how to deal with the qualification of toolchains. As toolchains are becoming increasingly complex and highly integrated in modern software development and verification environments (Wildmoser *et al.,* 2012), there is a need for further discussion on approaches, such as the ones described herein, for properly qualify software toolchains.

Another approach proposed to demonstrate tool confidence is based on verifying whether a tool behaves as expected upon a fault injection. Wang *et al.* (2012) presented in their study a semi-automatic method for achieving verification tool qualification in accordance with ISO 26262 tool confidence requirements. The qualification method proposed in that study consists in the application of functional stimuli and verifying if it is detected by the tool under analysis. In case all faults are detected, the qualification is complete. If a mismatch in the results is identified, corrective action shall be taken. This method is mainly proposed to reduce the effort for re-qualification of modified verification tools. It discusses one of the challenges for achieving proper compliance with qualification requirements, that is the effort required to qualify tools that need to be frequently modified.

Silva *et al.* (2019) also described a fault injection approach, presenting a methodology for functional safety verification in the context of compliance with ISO26262. This methodology consists in automating the execution of automatic test generators, formal methods analysis, and fault injection simulation. Once the execution of these three tools is concluded, their results are compared to verify potential mismatches among them. The adoption of redundant tools, as presented in that study, is a possible method to detect malfunctions in the tools, thus supporting tool qualification efforts.

Other studies also address different proposed methods or approaches for tool qualification. Prabhu *et al.* (2018) study discusses the development of a full-authority digital engine control (FADEC), addressing how a few safety-critical standards aspects are considered in the software life cycle. The study briefly addresses the method applied to gain confidence on the output of a source code auto-generation tool. To gain confidence on the output produced by the in-house source code generator, the proposed methodology was to check both source codes with different verification tools (source code generated with in-house developed tool was verified with LDRA Test tool, while MathWorks source code generated was verified with RTRT Test Tool) and compare its results. If the results match, the confidence on the outputs generated by the in-house tool would be demonstrated.

Wagner *et al.* (2017) study presents an approach of using a second, qualified tool to verify the outputs of the other unqualified tool. Hughes *et al.* (2022) study describes a method proposed for validation of digital forensic tool (a file carving tool) through the application of a continuous measurement process that provides for the gathering of enough data results to allow statistics analysis. Horsman (2019) study also discusses the need and the obstacles for implementing a tool-testing approach for digital forensic tools that would provide for greater confidence that the digital forensic tools outputs are reliable. The study identifies areas or aspects of the digital forensic tools that requires some level of tool-testing to ensure the correctness of the output.

In Guo *et al.* (2009) paper it is discussed the increasing need to ensure electronic evidence reliability, through the validation and verification of the tools and processes used to generate such evidence. The study proposes a functionality-oriented approach for validation of digital tools, starting from the modeling of the electronic evidence field activities, followed by the mapping of the relevant functions necessary to execute the procedures identified in the broader electronic evidence investigation model. The proposed approach follows with the definition of requirements for the functions mapped and their verification. The study advocates that the proposed approach provides for the possibility of qualifying small portions of the tool in a segregated way, can be extensible (specification can be added to the existing ones), tool agnostic approach, since the requirements, the set of test cases and expected results are based on the functions, does not matter the tool which will be subject to this validation.

Two of the selected studies presented proposed tool qualification approaches based on the application of a development process while developing the software tool. Trompouki and Kosmidis (2019) study presented an approach proposed for qualification of a toolchain utilized in the development of software for automotive systems. The study discusses the qualification, per ISO 26262 tool qualification requirements of an existing toolchain. Due to the size of the toolchain, only a few features of the tool were subject to this qualification process. The paper proposed a method for qualification, which was to develop the tool following some development process. The paper identifies the Automotive SPICE as the development process applied.

Durak *et al.* (2020) study discusses the applicability of tool qualification aspects for simulation-based system engineering. The study presents an overview of the IEEE recommended practice for Distributed Simulation Engineering and Execution Process (DSEEP), which defines processes to develop and execute simulations, drawn a comparison with DO-330 and proposes its use as basis for tool qualification.

Some relevant findings may be learned from the studies assessed to respond these research questions RQ1 and RQ2. Firstly, there is a well identified lack of guidance in existing safety standards on how to address the concerns around the use of complex and integrated toolchains. As well described in Asplund *et al.* (2012), the existent safety standards address tool qualification in a bottom-up approach, focused on individual, isolated tools. In a toolchain scenario, the possibility of a tool introduces errors or be unable to detect them may be caused not only by defects in the tool itself, but also by the integration of multiple tools. For that reason, aiming to address top-down integration issues, a system approach, benefiting from existent methodologies as HAZOP and STPA, are proposed.

A second notably finding observed from the studies assessed is that most of the studies proposing alternative approaches are those addressing Automotive domain, for which compliance with ISO 26262 is required. That may be explained by a less prescriptive nature of ISO 26262 in comparison with DO-330. According to ISO 26262, once the Tool Confidence Level required is defined, based on the ASIL (Automotive Safety Integrity Level) will be determined the required qualification methods. Among the possible applicable qualification methods identified in the standard, the “validation of the software tool” allows for approaches other than developing a tool in accordance with a safety standard.

4.2 RQ3 and RQ4 – Activities and Artifacts for Tool Qualification

The purpose of these research questions RQ3 and RQ4 was to identify activities performed and artifacts generated for achieving tool qualification. One aspect discussed in some of the studies assessed is related to the reuse of artifacts from a previous qualification to another. This issue is of particular importance for commercial of-the-shelf (COTS) tools, which are developed once and used in multiple projects and domains. Marques and Cunha (2017) addressed this concern, presenting an overview of the DO-330 tool qualification standard and discussed some pitfalls that are commonly identified when tool users pursue the qualification of commercial-of-the-shelf (COTS) tools. The study discusses the activities and artifacts produced for qualification of a TQL-5 COTS software tool, as well as some challenges to properly achieve tool qualification requirements. For that, Tool Operational Requirements and its corresponding verification data are required to be produced. For COTS tools, their vendor usually provides Tool Qualification Kits containing, in the case of TQL-5 tools, the Operational Requirements and their verification cases and procedures, to be executed by the user in its environment to generate the results needed for the qualification.

Slotosch (2014) study presents a model-driven approach to achieve tool qualification, that is shown to address DO-330 objectives. The model-based approach proposed provides, as claimed by the study, clarity on tool qualification elements, reusability, completeness, and automatization of the tool qualification tasks. The study provides a view on the application of model-based technique to make tool qualification effort less costly and risky.

Gallina et al. (2014) studies present an approach that intends to allow the reuse of tool qualification artifacts from one domain to another. The paper presents an approach to achieve tool qualification, through the reuse of tool qualification artifacts from other domains. This is based on the definition of a tool qualification process line and the use of process-based argument, using GSN (Goal Structuring Notation) for documenting safety cases.

Concerning specific activities conducted to ensure confidence in tool functionality, Marques and Cunha (2017) paper discuss the tailoring of DO-330 safety standard to attend a specific domain, in that case aeronautical databases, particular needs. The study addresses the use of qualified tools to either generate or verify the correctness of a Database File.

Tietz et al. (2021) study proposes a framework that supports the application of domain-specific modeling in safety-critical applications, with focus on streamlining qualification and automation of processes. The proposed framework addresses aspects of qualification, identifying activities to be performed towards ensuring domain-specific modelling tool confidence, which includes tasks such as making definitions on the meta- language used for modelling, ensuring the implementation of the model uses a proved and restrictive programming language, ensuring the model transformations outputs are generated in a deterministic and traceable way, as well as supports qualification artifacts generation.

Blackburn and Busser (1996) study describes the capabilities and the use of the tool T-VEC in the development and verification of two avionics systems. Among several different tool's modules, some of them were identified as subject to tool qualification, such as compiler, generator of test vectors, coverage analyzer and test driver's generator. Verification and coverage analysis processes were conducted, in a more extensive way, for the test vector generator module, to demonstrate that the obtained results matched the expected ones. Batista and Monsuez (2020) study discusses how to integrate artificial intelligence applications in system engineering environment, while in compliance with DO-330. The main goal of the study is to propose an architectural arrangement for Model Based System engineering tooling that could benefit from Artificial Intelligence capabilities to address issues present in the current MBSE tools, specially concerning the need for laborious, error prone data transformation in existing MBSE tooling.

Tian and Wang (2011) study provides a brief visibility on the usage of software tools in the context of airborne electronic hardware development and verification, in compliance with DO-254 standard. The study highlights the definitions and criteria established in DO-254 to determine the need for tool qualification. Kornecki and Zalewski (2010) study describes tool qualification efforts in the context of Programmable Logic Devices (PLD). The study discusses the tool assessment criteria as defined in DO-254 and provide visibility in the most common set of activities to ensure confidence in the tool: review of the outputs; repeat a subset of the simulation verification cases in the actual target hardware and compare the results; perform the same life cycle activity with another, alternative tool and compare the results.

Wagner *et al.* (2017) study presents the results of a case study conducted on formal methods tools qualification. The study describes two case studies, the first describing the activities and artifacts necessary to qualify Kind 2 model Checker tool per TQL-5 DO-330 objectives. As required per DO-330, Tool Operational Requirements were specified, test procedures developed and executed. Borgh and Towhidnejad (2019) presented in their study some guidelines for developing tool operational requirements for model checking tools. The study identified the need of appropriate specification of tool operational requirements for successful qualification of verification tools. Among the studies addressing activities performed and the artifacts generated to demonstrate compliance with tool qualification requirements or objectives, such as those defined in safety standards like DO-178C/DO-330 and ISO 26262, the issue concerning the reuse of artifacts for compliance with safety standards is of special importance, being pointed out that the possibility of reusing safety standards artifacts would contribute to a more lean, less costly and risky tool qualification effort (Gallina *et al.,* 2014; Marques and Cunha, 2017; Slotosch, 2014; Taft *et al.,* 2017).

4.3 RQ5 – Challenges to Tool Qualification

The purpose of this research question RQ5 was to verify the challenges to demonstrate compliance with tool qualification requirements or objectives. Several of the selected studies discussed challenges faced by practitioners when conducting tool qualification efforts. Notably, the increased utilization of bigger and complex toolchain throughout the safety-critical software life cycle, which according to Baumgart et al. (2021) is becoming increasingly common in modern software developments, is the most cited challenge to achieve proper and efficient tool qualification. In an analogous way, the utilization of open- source tools also impose new challenges since these tools may not be able to address and provide all needed evidence for compliance with the safety standards.

Baumgart et al. (2021) study discusses some of these concerns, by describing a qualification approach for a chain of tools. In their study, they claim that the existing guidance for tool qualification in different safety standards, such as ISO 26262 and DO-178C, are applicable for single tools, but are not adequate for a toolchain. This study identifies the utilization of complex toolchains as a challenge to achieve tool qualification requirements. The study highlights that manually managing a complex toolchain and keeping tracking of its configuration is time consuming and error prone.

De la Vara *et al.* (2021) work describes the development of an open-source environment for the AMASS tool, which is intended to support the certification and assurance of cyber- physical systems. Although a specific discussion on the qualification of the AMASS tool is not addressed, it is noted that the integration of different tools, in a toolchain, as well as the usage and needs when applying the tool in different domains plays a relevant role in this context of tool qualification.

Taft and Bordin (2014) study describes a few challenges to be addressed in tool qualification process. The study describes the different criteria for tool qualification in multiple safety standards as one challenge for tool qualification, since COTS tools may be used in different domains requiring qualification. Lack of proper guidance for qualification of software toolchains is also identified as an important topic for properly demonstrate compliance with tool qualification. The study also identifies the lack of an aligned approach for qualification of formal methods tools as a challenge to be overcome in tool qualification.

Two Kornecki and Zalewski (2003, 2008) studies present the results of a survey where respondents from the industry identified the major difficulties faced to qualify tools. Among the difficulties identified, the rigor on the required documentation for COTS tools was considered as making it impractical to qualify. The survey also identified that the reuse of tool qualification data would greatly contribute to easy tool qualification efforts. Kornecki *et al.* (2004) study provide visibility on an experiment conducted to measure the impact due to the use of tools while developing a safety-critical application. Notably, the experiment identified an increased amount of time spend by software development team to learn how to configure and operate the tools, when it is first introduced. The tool learning curve effort was later compensated by a more efficient code generation, which allowed the project to finish on time.

Schuster (2022) provides a description on the need for tool qualification, emphasizing the extensive use of software tools in the life cycle of safety-critical software. The study identifies a few challenges to be accounted for throughout tool qualification effort, such as the lack of proper tool documentation, mishandling of tool version or configuration data, inadequate assignment of required tool qualification level, insufficient tool operational requirements and testing. Tietz *et al.* (2022) study presents a discussion on the use of domain- specific modelling (DSM) tools in the development and verification of avionics software. The paper identifies as one of the means to ensure the correctness of the outputs is through the qualification of the modelling tool. The study identifies challenges to be overcome to qualify DSM tools, such as the fact that these DSM tools are implemented using Object Oriented Programming Languages, which may impose difficulties to ensure the outputs are produced in deterministic way. Another challenge, difficult to achieve compliance with the applicable tool qualification level is the lack of documentation of these tools, as well as the high complexity of these DSM tools, which includes multiple functionalities, transformation, and creation of models, adding for making it difficult to ensure the outputs will be produced deterministically.

Another challenge for achieving tool qualification identified in the studies assessed is regarding the qualification of formal methods tools, since compliance with a safety standard such DO-330 may be a challenge, due to the need to ensure the tool behaves as expected by the specified tool requirements. Bhatt *et al.* (1996) study discusses the opportunities and difficulties faced by practitioners to utilize formal methods tools in avionics software. This study identified that lack of documentation and difficult to completely verify the tool behavior are challenges to demonstrate compliance with DO- 330 objectives. The study outlines some strategies to overcome these obstacles.

Aiello *et al.* (2020) study proposes an assurance-case based on overarching properties approach for tool qualification, to mitigate or circumventing the challenges imposed by the DO-330 objectives when applied to the qualification of a code generator tool. Cofer (2015) work also discusses the challenges in the qualification of formal methods tools. The study present that although DO-178C, DO-330 and DO-333 provide guidance on both tool qualification and formal methods, there are still many issues and uncertainties while a practitioner conducts a qualification effort for formal methods tools. The paper identifies that not only the qualification of the formal methods tool is a challenge, but it is also a question for the translators used to convert an artifact to the input language of the formal method tool. The study indicates the many advantages in using formal methods tools, by reducing the effort in verification and identifying issues earlier but highlights the challenge to achieve proper tool qualification for these formal methods tools.

Other concerns and challenges identified in the studies assessed are related with the support needed to properly address the safety standards requirements. Kraus *et al.* (2015) study argues that tools used in safety analysis should also be subject to tool qualification requirements. Although DO-330, for instance, is intended for any domain, adapting the context, criteria and artifacts generated to address the applicable requirements of this standard for a safety analysis tool would be a challenge.

5 Conclusion

The purpose of this SLR was to evaluate the existing literature discussing software tool qualification in the context of safety- critical applications to identify approaches and/or methods employed to ensure confidence in tools functionality, as well as its activities and artifacts produced to substantiate the qualification claim. It was also the purpose of this SLR investigate the challenges to properly achieve and demonstrate tool's functionality that allow certification credit to be taken. The following conclusions may be acquired from the review, analysis and discussion performed.

Approaches and Methods for Tool Qualification. There are several studies, especially those concerned with Automotive domain, discussing proposed methods to demonstrate that a tool, or a collection of tools integrated in a so-called toolchain, produces correct and reliable outputs, so that credit from the outcomes of these tools may be granted without further confirmation. Some of the methods proposed for confirming tool confidence apply a model-driven approach, relying on modelling techniques to identify portions of tools deemed more relevant and, thus, subject to tool qualification. Other proposed methods used a more systemic approach, utilizing system-level safety analysis, such as STPA and HAZOP. These studies argue that the evaluation of the confidence of a toolchain should focus on the entire toolchain, rather than in each portion of it, so that any possible hazard that may be caused by the interactions among different parts of the toolchain are properly mitigated. Other possible approaches are also identified, such as fault- injection and the application of development processes. From the assessment of the studies, it became clear that there is a lack of guidance concerning qualification of complex and integrated software toolchains. The current guidance provided by the existent safety standards does not provide for addressing issues that may appear with the integration of multiple tools in complex chains.

Activities, artifacts, and challenges in tool qualification. The discussion of activities performed, and artifacts generated for tool qualification are mostly related to the studies that discusses methods to demonstrate tool confidence. That seems natural, since directly associated with a method are the outcomes it must produce to properly demonstrate its purpose was achieved. The lack of proper guidance in the current safety standards to address the issues raised when utilizing big and complex toolchains is the most cited challenge in the studies selected. The challenges imposed by the utilization of toolchains in safety-critical applications becomes of the particular concern when modern development and verification environments are becoming increasingly integrated and complex, a trend for improving software development and verification efficiency. The burden imposed to practitioners due to a lack of guidance to allow or at least optimize the reuse of tool qualification data from one domain to another is also mentioned as an important challenge, even more important when considering scenarios of more complex toolchains integrating open-sourced tools.

The challenges involved with the qualification of formal methods software tools are also a critical point of concern in the studies. Addressing DO-330 objectives is seeing as a challenge for formal methods tools, due to the need to verify the tool behavior.

Acknowledgements

The authors wish to thank São Paulo Research Foundation (FAPESP). This work was supported in part by a grant from 2023/03393-5.

References

Aiello, M., Comar, C., & Ruiz, J. (2020). An Assurance Case based on Overarching Properties for a TQL1 Code Generator. *Embedded Real Time Systems ERTS 2020.* IEEE Xplore.

Asplund, F., Biehl, M., & Loiret, F. (2012). Towards the Automated Qualification of Tool Chain Design. *Computer Safety, Reliability, and Security. SAFECOMP 2012.*

Asplund, F., El-khoury, J., & Törngren, M. (2012). Qualifying Software Tools, a Systems Approach. *Computer Safety, Reliability, and Security. SAFECOMP 2012.*

Batista, L., & Monsuez, B. (2020). The conception of a large-scale Systems Engineering environment. *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC).* San Antonio, TX, USA: IEEE Xplore.

Baumgart, S., Chen, Y., Hamrén, R., & Punnekkat, S. (2021). A Model-Based Approach to Document Software Toolchains for Supporting a Safety Analysis. *2021 IEEE International Systems Conference (SysCon).* Vancouver, BC, Canada: IEEE XPlore.

Blackburn, M., & Busser, R. (1996). T-VEC: a tool for developing critical systems. *Proceedings of 11th Annual Conference on Computer Assurance. COMPASS '96.* Gaithersburg, MD, USA: IEEE Xplore.

Blackburn, M., & Busser, R. (1996). T-VEC: a tool for developing critical systems. *Proceedings of 11th Annual Conference on Computer Assurance. COMPASS '96.* Gaithersburg, MD, USA: IEEE Xplore.

Borgh, A., & Towhidnejad, M. (2019). Guidelines for Development of Operational Requirements for Model Checking Tools. *2019 Integrated Communications, Navigation and Surveillance Conference (ICNS).* Herndon, VA, USA: IEEE Xplore.

Brauer, J., Peleska, J., & Schulze, U. (2012). Efficient and Trustworthy Tool Qualification for Model-Based Testing Tools. *Testing Software and Systems. ICTSS 2012.*

Cofer, D. (2015). You keep using that word. *ACM SIGLOG News 2, 4 (October 2015)*, 17-25.

da Silva, F., Bagbaba, A., Hamdioui, S., & Sauer, C. (2019). Efficient Methodology for ISO26262 Functional Safety Verification. *2019 IEEE 25th International Symposium on On-Line Testing and Robust System Design (IOLTS).* Rhodes, Greece: IEEE Xplore.

de la Vara, J., Ruiz, A., & Blondelle, G. (2021). Assurance and certification of cyber–physical systems: The AMASS open source ecosystem. *Journal of Systems and Software*.

Durak, U., D'Ambrogio, A., & Bocciarelli, P. (2020). Safety-critical simulation engineering. *In Proceedings of the 2020 Summer Simulation Conference (SummerSim '20).* San Diego, CA, USA: ACM Digital Library.

Gallina, B., Kashiyarandi, S., Zugsbratl, K., & Geven, A. (2014). Enabling Cross-Domain Reuse of Tool Qualification Certification Artefacts. *Computer Safety, Reliability, and Security. SAFECOMP 2014.*

Gleirscher, M., Sachtleben, R., & Peleska, J. (2023). Qualification of proof assistants, checkers, and generators: Where are we and what next? *Science of Computer Programming.*

Guo, Y., & Slay, J. (2010). A Function Oriented Methodology to Validate and Verify Forensic Copy Function of Digital Forensic Tools. *2010 International Conference on Availability, Reliability and Security.* Krakow, Poland: IEEE Xplore.

Guo, Y., Slay, J., & Beckett, J. (2009). Validation and verification of computer forensic software tools— Searching Function. *Digital Investigation*, S12-S22.

Horsman, G. (2019). Tool testing and reliability issues in the field of digital forensics. *Digital Investigation*, 163-175.

Hughes, N., Rabieh, K., & Aydogan, A. (2019). Employing a Continuous Measurement Process During Digital Tool Validation. *2019 International Symposium on*

Networks, Computers and Communications (ISNCC). Istanbul, Türkiye: IEEE Xplore.

Kornecki, A., & Zalewski, J. (2003). Design tool assessment for safety-critical software development. *28th Annual NASA Goddard Software Engineering Workshop, 2003. Proceedings.* Greenbelt, MD, USA: IEEE Xplore.

Kornecki, A., & Zalewski, J. (2008). Software certification for safety-critical systems: A status report. *2008 International Multiconference on Computer Science and Information Technology.* Wisla, Poland: IEEE Xplore.

Kornecki, A., & Zalewski, J. (2010). Hardware certification for real-time safety-critical systems: State of the art. *Annual Reviews in Control*, 163-174.

Kornecki, A., Hall, K., Hearn, D., Lau, H., & Zalewski, J. (2004). Evaluation of software development tools for high assurance safety critical systems. *Eighth IEEE International Symposium on High Assurance Systems Engineering, 2004. Proceedings.* Tampa, FL, USA: IEEE Xplore.

Krauss, S., Rejzek, M., & Hilbes, C. (2015). Tool Qualification Considerations for Tools Supporting STPA. *Procedia Engineering*, 15-24.

Marques, J., & da Cunha, A. (2015). Use of the RTCA DO-330 in aeronautical databases. *2015 IEEE/AIAA 34th Digital Avionics Systems Conference (DASC).* Prague, Czech Republic.

Marques, J., & da Cunha, A. (2017). COTS tool qualification using RTCA DO-330: Common pitfalls. *2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC).* St. Petersburg, FL, USA: IEEE Xplore.

Marques, J., & da Cunha, A. (2017). Verification scenarios of onboard databases under the RTCA DO- 178C and the RTCA DO-200B. 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC). St. Petersburg, FL, USA: IEEE Xplore.

Mishra, A., Rao, M., CU, C., Rao, V., Jeppu, Y., & Murthy, N. (2013). An auto-review tool for model-based testing of safety-critical systems. In Proceedings of the 2013 International Workshop on Joining AcadeMiA and Industry Contributions to testing Automation (JAMAICA 2013). New York, NY, USA: ACM Digital Library.

Prabhu, S., Kapil, H., & Lakshmaiah, S. (2018). Safety Critical Embedded Software: Significance and Approach to Reliability. 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI). Bengaluru, India: IEEE Xplore.

Schuster, G. (2022). Certification of software tools used in safety-critical software development. IEEE 5th International Conference and Workshop Óbuda on Electrical and Power Engineering (CANDO-EPE). Budapest, Hungary: IEEE Xplore.

Slotosch, O. (2014). Model-Based Tool Qualification. Information Technology and Open Source: Applications for Education, Innovation, and Sustainability.

Taft, S., Richa, E., & Toom, A. (2017). Building Trust in a Model-Based Automatic Code Generator. ACM Digital Library.

Taft, T., & Bordin, M. (2014). Towards a lean tool qualification process: Digital avionics systems conference. 2014 IEEE/AIAA 33rd Digital Avionics Systems Conference (DASC). Colorado Springs, CO, USA: IEEE Xplore.

Tian, Y., & Wang, P. (2011). Research of software tools for DO-254 projects. 2011 International Conference on Computer Science and Service System (CSSS). Nanjing: IEEE Xplore.

Tietz, V., Frey, C., Schoepf, J., & Annighoefer, B. (2022). Why the use of domain-specific modeling in airworthy software requires new methods and how these might look like? In Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings (MODELS '22). ACM Digital Library.

Tietz, V., Schoepf, J., Waldvogel, A., & Annighoefer, B. (2021). A Concept for a Qualifiable (Meta)- Modeling Framework Deployable in Systems and Tools of Safety-Critical and Cyber-Physical Environments. 2021 ACM/IEEE 24th International Conference on Model Driven Engineering Languages and Systems (MODELS). Fukuoka, Japan: IEEE Xplore.

Trompouki, M., & Kosmidis, L. (2019). BRASIL: A High-Integrity GPGPU Toolchain for Automotive Systems. 2019 IEEE 37th International Conference on Computer Design (ICCD). Abu Dhabi, United Arab Emirates: IEEE Xplore.

Wagner, L., Mebsout, A., Tinelli, C., Cofer, D., & Slind, K. (2017). Qualification of a Model Checker for Avionics Software Verification. NASA Formal Methods. NFM 2017.

Waldvogel, A. (2022). Towards qualifiable graphical editing of complex domain-specific models in safety- critical avionics. In Proceedings of the 25th International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings (MODELS '22). New York, NY, USA: ACM Digital Library.

Wang, Q., Wallin, A., Izosimov, V., Ingelsson, U., & Peng, Z. (2012). Test tool qualification through fault injection. 2012 17th IEEE European Test Symposium (ETS). Annecy, France: IEEE Xplore.

Wildmoser, M., Philipps, J., & Slotosch, O. (2012). Determining Potential Errors in Tool Chains. Computer Safety, Reliability, and Security. SAFECOMP 2012. Springer Link.

Additional papers used as reference:

Biolchini, J., Mian, P., Natali, A., & Travassos, G. (2005). Systematic Review in Software Engineering. Tech. Rep. RT-ES 679/05, COOPE/UFRJ,, vol. 107, no. 1273 SPEC. ISS., pp. 32–37.

Boulanger, J. (2015). CENELEC 50128 and IEC 62279 Standards. Wiley, 2015. Wiley.

Camus, J.-L.E. (2014). Tool qualification in multiple domains: Status and perspectives. Embedded real-time software and systems (ERTS² 2014).

Federal Aviation Administration. (2017). Advisory Circular 20-115D - Airborne Software Development Assurance Using EUROCAE ED-12( ) and RTCA DO-178( ). 2017. FAA.

Federal Aviation Administration. (2018). Order 8110.49A - Software Approval Guidelines. 2018. FAA.

Food and Drugs Administration. (2024, March 23). "Recognized Consensus Standards," Date of Entry 01/14/2019. (accessed March 2024). Retrieved from https://www.acessdata.fda.gov/scripts/cdrh/cfdocs/cfstandards/detail.cfm?standard dentification_no=38829

International Electrotechnical Commission. (2015a). IEC 62279:2015 Railway Applications - Communication, Signalling And Processing Systems - Software For Railway Control And Protection Systems. IEC, 2015. IEC.

International Electrotechnical Commission. (2015b). IEC 62304:2006/AMD 1:2015 Medical Device software — Software life cycle processes — Amendment 1. IEC, 2015. IEC.

International Standardization Organization. (2018). ISO 26262-6:2018 Road vehicles — Functional safety— Part 6: Product development at the software level. ISO, 2018. ISO.

Kitchenham, B., & Charters, S. (2007). Guidelines for performing Systematic Literature Reviews in Software Engineering. Version 2.3 EBSE Tech. Rep. EBSE- 2007-01, 2007, doi: 10.1109/ACCESS.2016.2603219.

Knight, J. (2002). Safety critical systems: challenges and directions. Proceedings of the 24th International Conference on Software Engineering. ICSE 2002, (pp. 547-550).

Martins, L., & Gorschek, T. (2016). Requirements engineering for safety-critical systems: A systematic literature review," Inf. Softw. Technol., vol. 75, pp. 71–89, 2016, doi: 10.1016/j.infsof.2016.04.002. Inf. Softw. Technol., vol. 75, 71-89.

RTCA Inc. (2011a). DO-331, Model-Based Development and Verification Supplement to DO- 178C and DO-278A. RTCA.

RTCA Inc. (2011b). DO-332 - Object Oriented Technology and Related Techniques Supplement to DO-178C and DO-278A. RTCA.

RTCA Inc. (2011c). DO-333 - Formal Methods Supplement to DO-178C and DO278A. RTCA.

RTCA, Inc. (1992). DO-178B - Software Considerations in Airborne Systems and Equipment Certification. 1992. RTCA.

RTCA, Inc. (2011d). DO-178C – Software Considerations in Airborne Systems and Equipment Certification. 2011. RTCA.

RTCA, Inc. (2011e). DO-248C, Supporting Information for DO-178C and DO-278A. RTCA Inc., 2011. RTCA.

RTCA, Inc. (2011f). DO-330 - Software Tool Qualification Guidelines. 2011. RTCA.

RTCA, Inc. (2016). DO-200B - Standards for Processing Aeronautical Data RTCA, 2016. RTCA.

Rushby, J. (1994). Critical system properties: survey and taxonomy. Reliability engineering & system safety, Vol.43(2), pp.189-219, doi: https://doi.org/10.1016/0951-8320(94)90065-5.

SAE International. (2012). ARP-4754 A -- Guidelines for Development of Civil Aircraft and Systems. SAE International, 2012. SAE International.