STUDY ON TECHNICAL FOCUSES AND SAMPLING COVERAGE STRATEGY OF AIRBORNE SOFTWARE REVIEWS

Jinghua Sun[1], Samuel Edwards[2], Nic Connelly[3], Andrew Bridge[4] and Lei Zhang[1]

[1]COMAC Shanghai Aircraft Design and Research Institute, Shanghai, China
[2]Defence Aviation Safety Authority, 661 Bourke St, Melbourne, VIC, Australia
[3]School of Engineering, RMIT University, Melbourne, VIC, Australia
[4]European Union Aviation Safety Agency, Cologne, Germany

ABSTRACT

Airborne software is invisible and intangible, it can have a significant impact on the safety of the aircraft. However, it cannot be exhaustively tested, and can only be assured through a structured, process, activity, and objective-based approach. This paper studied the similarities and differences of software review policies of the four selected National Airworthiness Authorities (NAAs) by using a comparative approach and analysed the general certification basis of specific regulation clauses from the International Civil Aviation Organization Conventions to each contracting States' regulations by a traceability method. Then analyzed the development processes and objectives applicable to different software levels based on RTCA/DO-178C. Identified 82 technical focus points based on each airborne software development sub-process, then created a Process Technology Coverage matrix to demonstrate the technical focuses of each process. Developed an objective-oriented top-down and bottom-up sampling strategy for the 4 software Stage of Involvement reviews by taking into account the frequency and depth of involvement. Finally, created the Technology Objective Coverage matrix, which can support the reviewers to perform the efficient risk-based SOI reviews by considering the identified technical points, thus to ensure the safety of the aircraft from the software assurance perspective.

KEYWORDS

Airborne Software, SOI, DO-178C, Objective, Sampling Strategy.

1. INTRODUCTION

With the development of computer technology, more and more aircraft system functions are implemented by airborne software. Almost all the civil aircraft systems, for instance, the flight control system, landing gear system, hydraulic system, fuel system, communication system, navigation system, display system, electrical power system, and power plant system, include software. The integration and complexity of software are increasing day by day. However, Software is an intangible asset, having no physical presence, which is stored on a variety of media (CASA, 2014). The software will fail only when there is a potential error, virus, design error, or single event exception. Software design errors may exist for many years without manifesting or causing malfunctions. Thus quality should be built into the software and be reviewed by assuring the development and verification processes (CASA, 2014) (Rierson, 2013). For the leading airworthiness authorities in the world, airborne software is always one of the key concerns in the aircraft certification process (EASA, 2012).

As competent NAAs need to maintain expertise in a wide range of aircraft design technologies (Hilderman & Baghai, 2007), for example, maintain expertise in the design of aircraft structures, propulsion systems, aviation software, electromagnetic environmental effects, human factors, performance, and handling, electrical systems, navigation systems, and many others. This project is focused on the study of the airborne software domain. The NAAs' software review officers, Designated Engineering Representatives (DERs), and the certification engineers of an Organization Designation Authorization (ODA) or Design Organization Approval (DOA) need to be familiar with airborne software developing processes and understand the related technologies to be able to do deep and effective reviews, supporting them to judge whether the software complies with applicable airworthiness regulations (FAA, 2004).

1.1. Research questions

This study focuses on software review strategy and related software technologies that need to be maintained by the reviewers. The term of technology in this paper means the sum of techniques, skills, methods, and processes used in the production of airborne software and the accomplishment of compliance objectives. Airborne software-related technology can be divided into different layers. Airborne software is the first layer technology, the main life cycle processes is second, then the sub-process can be treated as the third layer, the output and technical points of each sub-process is next, and the specific technical aspects of each sub-process is the last layer of this study, such as deactivated code, partitioning and so on, which is regarded as a technology in this paper.

Therefore, the key research question is: What is the ideal technology coverage and airborne software sampling strategy of software reviews for an Airworthiness Authority? This question contains the following three aspects.

Firstly, installed software is a subset of aircraft systems and equipment and is reviewed and approved as an integral part of the certification of the parent equipment (CASA, 2014). NAA's software reviews are official activities, which are used to find compliance with regulations. The first sub-question is: What is airborne software regulatory requirements? That means the certification basis of airborne software should be established prior to the software reviews.

Secondly, the main purpose of software review is to ensure the safety of software products (Wetherholt & Penix, 2002). Reviewers not only pay attention to the development process but also pay attention to the completeness and correctness of software design data through sampling reviews. So they should understand the technologies used in the software development process and their impact on safety. There are many specific technologies are used, such as Requirements-Based Testing (RBT), Tool Qualification, Model-Based Development (MBD), Object-Oriented Technology (OOT), and so on. The software reviewers should be able to find defects and loopholes in the software life cycle process to ensure the systems hosting airborne software to operate at an acceptable safety level (Jimenez, et al., 2020), which leads to the second question: What technologies should be identified and focused on during software reviews?

Thirdly, a software project can last for several years depending on the criticality, complexity, and maturity of the systems and many other organizational factors (Delange, et al., 2015), a NAA's software review Level of Involvement (LOI) criteria and review strategy can support the NAA's resource management and have a good balance between review frequency and depth. Given that samplings are picked from numerous artifacts, an intelligent strategy can support an effective and efficient software review (Chen, et al., 2015). The last sub-question comes with: what is the sampling strategy of software reviews?

1.2. Research objectives

Based on the statement of the problem, this study aims to explore an 'ideal' airborne software review and sampling coverage strategy for a NAA provided that identifying technical focuses based on software life cycle processes. The objectives include:

a) Capturing airborne software regulatory requirements and policies.
b) Identifying the range of airborne software technologies based on the typical advanced widely-Used airborne software life cycle process.
c) Developing the software sampling review strategy that can support a NAA to do a sufficient effective software review by taking into account the frequency and depth of the reviews.

2. BACKGROUND

Modern transport aircraft can be developed to have a stable flight profile, but embedded software is used to control and optimize the flight of the aircraft (Romanski, 2001). Software safety is an increasingly prominent issue in today's aviation industry (Mendis, 2008). The aircraft systems can directly affect the safety of aircraft. However, the software is fundamentally different from the physical components installed on the aircraft. Continuous testing cannot guarantee that the software has the same reliability level as the physical components. The structural components of the aircraft can be tested to ensure that there are no design and manufacturing defects, whereas the Mean Time between Failures (MTBF) and programmed replacements do not apply to software components (CASA, 2014). The software embedded in these systems also has a direct impact on the safety of the aircraft and its occupants (Hilderman & Baghai, 2007). Employing software review technology can ensure that rigor has been applied during the applicant's design commensurate with the worst-case failure condition associated with airborne software (RTCA, 2011a).

Therefore, a certain level of assurance is required to have confidence in software to ensure aircraft safety. In October 2018 and March 2019, two Boeing 737MAX planes belonging to Indonesia Lion Air and Ethiopian Airlines crashed respectively, causing a total of 346 deaths, which was directly related to the design of the Maneuvering Characteristics Augmentation System (MCAS) and its flight control law software (COMMITTEE, 2020). It has been a great loss for Boeing, at the same time the FAA as the supervisor, also triggered a crisis of public trust. Wayne Rash stated that "As is the case where software controls hardware, there are ways things can go wrong either because something happened that wasn't anticipated, or because the response was wrong" (Rash, 2019). So what can be done to ensure that the software to be maintained at an acceptable level of safety?

As software is invisible and intangible, it is unlike the structural parts of an aircraft that can be measured and verified through tests and inspections (CASA, 2014). Due to the particularity of airborne software and the professionalism of software-related technologies, high requirements are placed on airborne software reviewers. However, the complexity and scale of software keep increasing as modern civil airplanes are getting more and more integrated and complex. Figure 1 shows the estimated airborne Software Lines of Code (SLOC) growth of Boeing and Airbus aircraft. According to Information is Beautiful, there are a total of 14 million lines of codes in Boeing 787 (Beautiful, 2015). Formulating a set of airborne software review strategies with related technical focuses is an important issue.

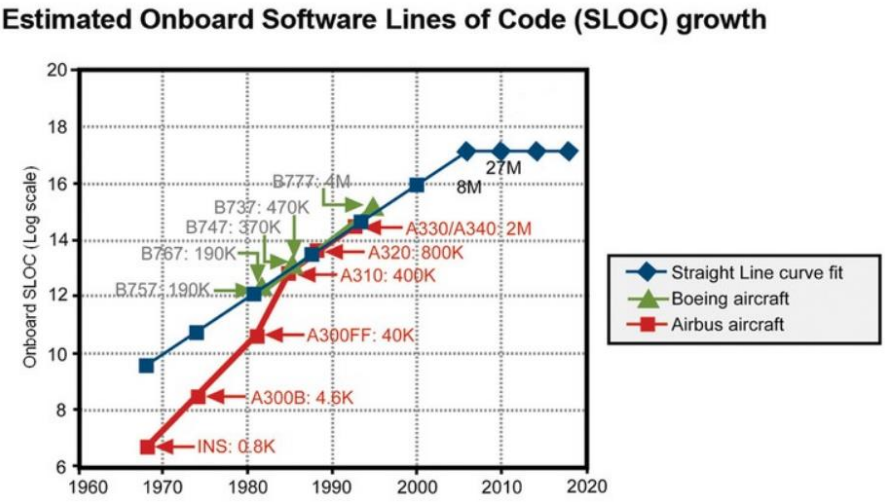

Figure 1. Illustration of Estimated SLOC Growing to Double Every Four Years
Source :(StClair & Hillary, 2012)

Given that software is but an executable computer file, it is not feasible to assess the number of kinds of software errors (FAA, 1988). For more than three decades, airborne software has been developed and assured through a structured approach based on objectives and activities (Rierson, 2013). The most commonly used method to measure software goodness is a document called, *Software Considerations in Airborne Systems and Equipment Certification*. In the US, it is called DO-178 and in Europe, it is called ED-12, which is recognized as a Means of Compliance (MOC) by NAAs via their respective Advisory Circular (AC) (Hilderman & Baghai, 2007). This study was carried out based on the latest version of the DO-178C software life cycle. The NAAs, Military Airworthiness Authorities (MAAs), software certification officers, DERs, ODAs, DOAs, and aircraft applicants will benefit from this study.

3. METHODOLOGY

3.1. A Step-by-Step Methodology

This project used a step-by-step methodology to do the research and solve the problem. It focused on the technology coverage for each SOI review based on the DO-178C life cycle. Firstly, it captured the airworthiness authority's regulatory requirements for software reviews in a comparative approach, and secondly studied the DO-178C life cycle process to identify the software review technical concerns and a comprehensive set of technologies related to each process, then formulated an ideal airborne software SOI review and sampling strategy by taking into account the identified technologies of each process, finally made a recommendation for improvement of software review guidance for the NAAs and a conclusion of the achievement of this research. The research route map was shown in Figure 2. The four-step approach research route map and technical solutions were carried out as follows.

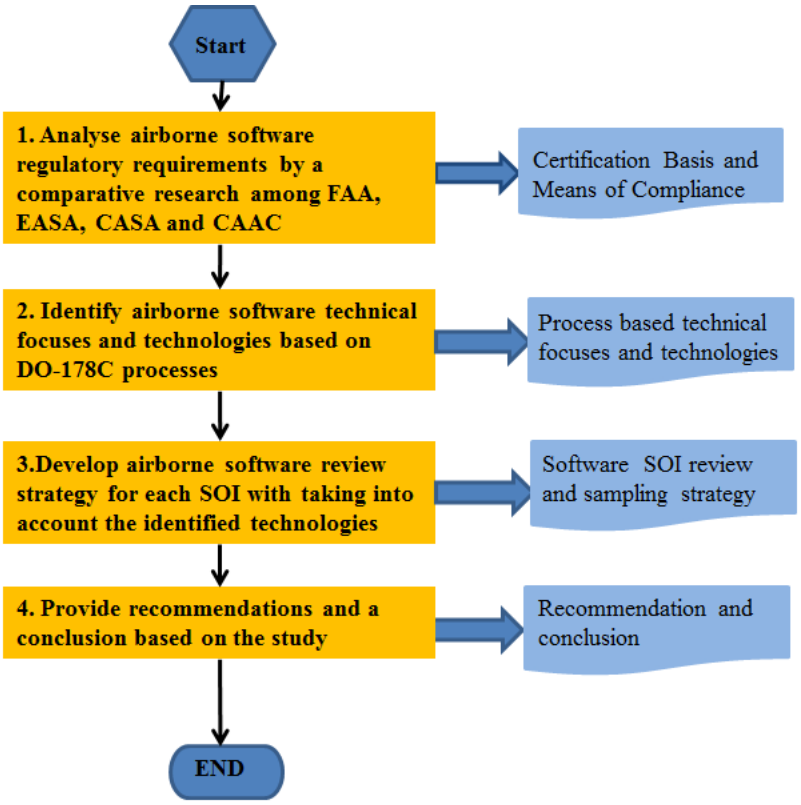

Figure 2. A step by step research methodology

Step1: Analyzed the airborne software certification basis and the MOC

According to the literature review, the airworthiness regulations of other Contracting States do not inherit ICAO's clear airworthiness requirements for airborne software in Appendix H of ICAO Annex 8. It is necessary to analyze the corresponding regulatory requirements for software to form the legal basis of software reviews. This analysis includes:

a) Capturing the requirements of airborne software from the ICAO convention as the root source.
b) Analyzing the traceability relationship between ICAO convention and each Contracting States' regulations.
c) Analyzing the clauses of the regulation that are applicable for airborne software to identify the common certification basis of airborne software.
d) Analyzing the airborne software compliance means recognized by NAAs and the justification.
e) Analyzing the software certification-related policies in a comparative approach among FAA, EASA, CASA, and CAAC.

Step 2: Identified the process based technical focuses

This paper used a quantitative research approach to analyze DO-178C processes and objectives according to the software Development Assurance Level (DAL), then identified technical focuses for each process, which mainly contains:

a) Analysis of DO-178C based software life cycle processes, the relationship between each process, and objectives applicable to different software DAL.
b) Analysis of Technical Focuses of DO-178C process.

Step 3: Developed the software SOI review strategy

Based on the DO-178C life cycle process and the identified technologies, this paper studied and formulated the review strategy for each SOI with a qualitative research approach. Aiming at the SOI#2 and SOI#3 software development and verification process, this paper created a sampling strategy to make the SOI reviews more efficient and effective. The following study is performed:

a) Based on the current documented LOI criteria generated by FAA and EASA, researched and supplemented several LOI criteria that are conducive to work practice.
b) Based on the analysis of SOI review procedures and manuals in FAA Order8110.40, EASA CM-AEHSW-002, and FAA Software Review Job Aid， developed a set of software review strategies combining software technical concerns and sampling strategies.
c) Quantitative analysis of process technologies and objectives applicable to each SOI review by a Process & Technology Coverage (PTC) matrix and a Technology & Objective Coverage (TOC) matrix.

Step4: Provided the recommendations and conclusions

Based on the above three-step study and analysis, this paper made recommendations on future improvement and summarized the achievement and limitation of this study.

3.2. Source of data

The sources of data are mainly from ICAO Conventions, the NAA's Regulations, industry standards, government official database including FAA, EASA and CAAC's websites, research papers, and reports from Google Scholar and RMIT e-library, and interviews of DASA software specialists, industry experts, and other certification specialists.

4. ANALYSIS OF SOFTWARE REVIEW TECHNICAL FOCUSES

4.1. Comparative Analysis of Airborne Software Regulatory Requirements

The leading NAAs recognize DO-178C as an acceptable MOC of airborne software through ACs (FAA, 2013) (EASA, 2013) (CASA, 2014). They also published many policies to guide the software review and approval.
Table 1 provides a comparison of current software review policies of FAA, EASA, CASA, and CAAC from the following dimensions:

a) The recognized MOC,
b) The main content comparison of software MOC ACs,
c) Software review and approval guidelines,

d) Delegation mechanism,
e) LOI criteria,
f) SOI review strategy, and
g) Data submittal requirements.

Table 1. The general comparison of the NAAs' software review policies
Source: (CASA, 2014) (FAA, 2017) (EASA, 2017) (CAAC, 2000) (EASA, 2012) (FAA, 2018) (Cai, 2020) (CASA, 2019) (CASA, 2011) (FAA, 2011)

Comparison Dimension	**FAA**	**EASA**	**CASA**	**CAAC**
The recognized MOC	DO-178C	DO-178C	DO-178C	DO-178B
AC	AC20-115D (2017) Cancelled AC20-115C (published in 2013)	AMC20-115D (2017) Cancelled AMC20-115C (published in 2013)	AC21-50 (2014) Software is considered to be an aeronautical product following the Civil Aviation Act 1988 (the Act).	AC-21-02(2000) An updated AC to recognize DO-178C is pending release.
Softw are Guide lines	FAA Order 8110.49 A	CM-SWAEH-002 Issue 01 Revision: 01	Refer to FAA Order 8110.49.	Refers to FAA Order 8110.49.
Deleg ation Mech anism	ODA and DER	DOA	Approved design organization (ADO) and Industry Delegates/Approved Persons	DER
LOI	Defined LOI criteria in FAA Order 8110.49 Chg1 Chapter 3 and Appendix A.	Defined LOI criteria in CM-SWAEH-002 Issue 01 Rev 01 in Chapter 5.	Refer to FAA Order8110.49 in AC 21-50.	No criteria have been released. CAAC's LOI is mainly based on a case-by-case approach.
SOI	Defined in Order 8110.49 A Chapter 2.	Defined in Chapter 4 of CM-SWAEH-002 Issue01 Rev01.	Refer to FAA Order8110.49.	Refer to FAA Order 8110.49. In CAAC internal manual for a civil aircraft certification program, besides 4 SOIs, they added another 2, which are System-Software Consistent Review before SOI#1 and Software maturity reviews

				depending on the project status.
Data Submittal	At least PSAC, SCI, and SAS. TQP and TAS if applicable.	At least PSAC, SCI, and SAS. TQP and TAS if applicable.	At least PSAC, SCI, SAS, and SQA records. TQP and TAS if applicable.	At least PSAC, SCI, and SAS. TQP and TAS if applicable.

Through comparative analysis, it can be known that the NAAs use the SOI review method and most of them have recognized DO-178C as an acceptable means of compliance of airborne software by ACs. Given that the differences mainly exist at their respective practical level, analyzing the DO-178C process and objectives allows identifying the technologies related to the software reviews.

4.2. Analysis of DO-178C Software Life Cycle Process and Objectives

DO-178C is a process-based activity-driven objective-oriented standard. It is not a software development standard, but a method to measure the goodness of software to ensure safety to be maintained at an acceptable level. It contains 6 processes (represented in Figure 3), which are planning process, development process, and 4 integral processes (verification process, configuration management process, quality assurance process, and certification liaison process). The integral processes are supported throughout the whole software lifecycle (RTCA, 2011a). It has to be mentioned, however, that not all the projects follow a perfect Waterfall model (Jimenez el. 2020), but a variation in the representation of the waterfall model instead (Santos and Ferreira 2019).

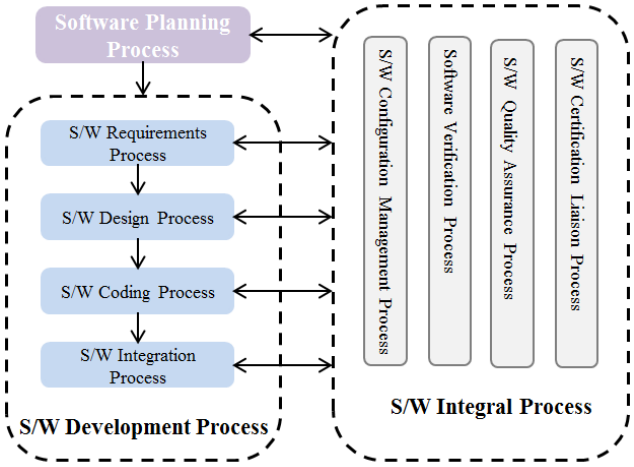

Figure 3. DO-178C software life cycle processes

A latent software error in data or the final product can cause a fault of the software, then the abnormal behaviors of software can lead to a system failure condition, which can finally affect the aircraft operations. The rigor of software development is determined by the software level. DO-178C defined 5 software levels as listed in Table 2, DAL A is the severest, while DAL E has no safety impact. The software DAL is determined by the system safety assessment process. The different level has different objectives requirements. Table 3 *and Figure 4* are the comparison of DO-178C's Objectives in Annex A from Table A-1 to Table A-10 for different DALs of software.

Table 2. DO-178C Software DAL , related failure conditions and objectives.
Source: (Marques & Yelisetty , 2019)

System Failure Condition	Required Software Level	Number of Associated Objectives	Number of Associated Objectives with Independence
Catastrophic	A	71	31
Hazardous	B	69	19
Major	C	62	5
Minor	D	26	2
No Safety Effect	E	0	0

Table 3. Comparison of DO-178C objectives for different software levels.

Annex A	A	B	C	D
Table A-1 Software Planning Process	7	7	7	2
Table A-2 Software Development Process	7	7	7	4
Table A-3 Verification of Outputs of Software Requirements Process	7	7	6	3
Table A-4 Verification of Outputs of the Software Design Process	13	13	9	1
Table A-5 Verification of Outputs of Software Coding & Integration Processes	9	9	8	1
Table A-6 Testing of Outputs of Software Integration Process	5	5	5	3
Table A-7 Verification of Verification Process Results	9	7	6	1
Table A-8 Software Configuration Management Process	6	6	6	6
Table A-9 Software Quality Assurance Process	5	5	5	2
Table A-10 Certification Liaison Process	3	3	3	3

The experience accumulation of reviewers can start from Level D software review, and gradually master the review methods and techniques of higher-level software, to finally be competent for the review of Level A software:

a) Level D can be treated as a black box, which only focuses on high-level requirements development and verification. If updating a level D software to level C, there will be a leap of workload.
b) The objectives differences between level C and level B include 1 Objective in Table A-3 “High-level requirements are compatible with target computer”, 4 objectives in Table A-4 about the compatibility and verifiability of low-level requirements and architecture, 1 objective in Table A-5 “Source code is verifiable”, and 1 objective about decision coverage in Table A-7.
c) The main differences between A and B are 2 objectives in Table A-7, which are requirements of MCDC Structural Coverage Analysis (SCA) and verification of additional code that cannot be traced to source code.

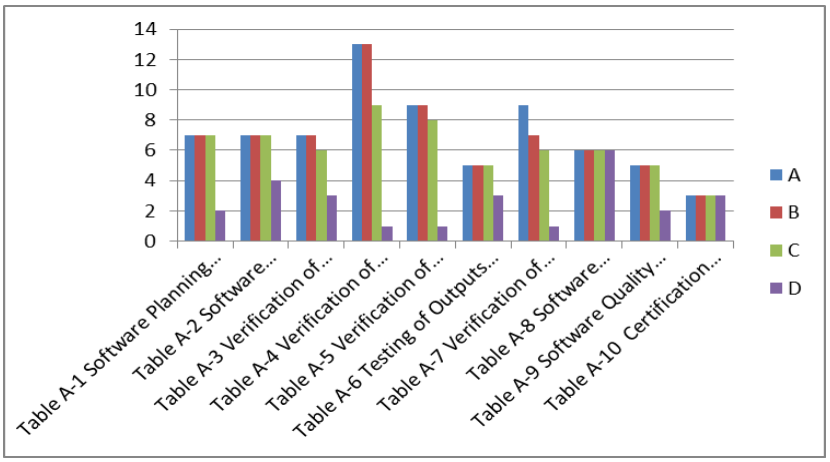

Figure 4. The comparison of applicable objectives in each Table of Annex A for different software levels

4.3. Analysis of Technical Focuses of DO-178C Process

Study on the DO-178C objectives and process can help software reviewers quickly locate the technical focuses and finding compliance. Table 4 is the analysis of the technologies based on DO-178C software life cycle processes. Each process of DO-178C may contain sub-process and components (RTCA, 2011a). The technical focus points are analysed based on each component, which should be covered and concerned during the software reviews. The technologies are from the research and analysis of the technical focus points with most of them are described in DO-178C and a few are from the industry practice, then they are compared with the CAST Paper research themes, finally re-analysed to ensure the completeness of the technology list.

Table 5. Qualitative analysis of technical focus points and related techniques of DO-178C
Source: (RTCA, 2011a) (FAA, 2003) (EASA, 2012) (FAA, 2004) (FAA, 2017) (CAST, 2002) (RTCA, 2011b)

Process	**Sub-process/ Components**	**Technical Focus Points/Elements**	**Technologies (T_i, i=1…n)**
4.0 Software Planning	4.3 Software Plans	11.1 PSAC 11.2 SDP 11.3 SVP 11.4 SCMP 11.5 SQAP	1) Software DAL Determination 2) Partitioning 3) Multiple-Version Dissimilar Software 4) Safety Monitoring 5) PDI 6) User-Modifiable Software 7) COTS 8) Field-Loadable Software 9) Option-Selectable Software 10) Software Life Cycle Definition 11) Transition Criteria 12) Deactivated Code 13) PDS 14) Tool Qualification 15) Reuse of tool qualification data 16) Reuse of software life cycle data 17) Exhaustive Input Testing 18) Software Reliability Model 19) Product Service History 20) Database/PDI 21) Use of COTS Graphical Processor Unit (GPU) 22) Microprocessor 23) Multiple Core Processors 24) SEU (Single Event Upset) 25) Reverse engineering
	4.4 Software Life cycle Environment Planning	4.4.1 Software Development Environment 4.4.2 Language and Complier 4.4.3 Software Test Environment	
	4.5 Software Development Standards	11.6 Software Requirements Standards 11.7 Software Design Standards 11.8 Software Code standards	
5.0 Software Development	5.1 Software Requirements	11.9 Software Requirements Data 11.22 Parameter Data Item File	26) High-Level Requirements 27) Derived requirements 28) Merging high-level requirements and low-level requirements
	5.2 Software Design	11.10 Design Description	29) Control Flow Design 30) Data Flow Design 31) Low-Level Requirements 32) PDI Design

Process	**Sub-process/ Components**	**Technical Focus Points/Elements**	**Technologies (T_i, i=1…n)**
	5.3 Software Coding	11.11 Source Code 11.22 Parameter Data Item File	33) C, Ada, Assembly languages 34) Auto code generation 35) MBD 36) OOT 37) Cache 38) Stack
	5.4 Integration	11.12 Executable Object Code	39) Compiling 40) Complier library 41) Software Integrity Check (e.g. Cyclic redundancy check, Checksum)
	5.5 Traceability	11.21 Trace Data	42) Traceability Tools (eg. DOORS)
6.0 Software Verification	6.3 Software review and analysis	Review and analysis of Software Plans and standards 6.3.1 Review and analysis of Software High-Level Requirements (HLRs) 6.3.2 Review and analysis of Software Low-Level Requirements (LLRs) 6.3.3 Review and analysis of Software Architecture 6.3.4 Review and analysis of Source Code 6.3.5 Review and analysis of the Outputs of the Integration Process 6.4.5 Review and analysis of Test Cases, procedures, and results 6.6 Review and analysis of PDI File	43) Plans and Standards Review 44) HLR Review and Analysis 45) LLR Review and Analysis 46) Architecture Review and Analysis 47) Source Code Review and Analysis 48) Outputs of the Integration Process Review and Analysis 49) Test Cases Review and Analysis 50) PDI file Review and Analysis 51) Worst-Case Execution Time 52) Verification of Stack Usage 53) Model Review and Analysis 54) Verification of independence
	6.4 Software Testing	6.4.1 Test Environment 6.4.2,6.2.3 Requirements-Based Test 6.4.4 Test coverage Analysis	55) Hardware/Software Integration Testing 56) Software Integration Testing 57) Low-Level Testing 58) Normal Range Test Cases Selection 59) Robustness Test Cases Selection 60) MCDC 61) Decision Coverage Analysis 62) Statement Coverage Analysis 63) Data Coupling 64) Control Coupling 65) DAL A additional verification (Whether

Process	**Sub-process/ Components**	**Technical Focus Points/Elements**	**Technologies (T_i, i=1…n)**
			Object Code can directly traceable to source code) 66) Extraneous Code Resolution 67) Deactivated Code Handle
	6.5 Traceability	11.21 Trace Data	
Integral Process	7.0 Software Configuration Management	7.2.1 Configuration Identification	68) Software part numbering
		7.2.2 Baselines and Traceability	69) Baseline Definition
		7.2.3 Problem Reporting	70) OPR Category Definition
		7.2.4 Change Control	71) Software Change Control
		7.2.5 Change Review	
		7.2.6 Configuration Status Accounting	
		7.2.7 Archive, Retrieval, and Release	72) Media Selection, Refreshing, Duplication 73) Data Retention
		7.3 Data Control Category	
		7.4 Software Load Control	74) Software Conformity Inspection
		7.5 Software Life Cycle Environment Control	
	8.0 Software Quality Assurance	8.2 Software Quality Assurance Activities	
		8.3 Software Conformity Review (SCR)	75) SCR 76) First Article Inspection (FAI)
	9.0 Certification Liaison	9.1 Means of Compliance and Planning (LOI, Milestones, and Issue Papers, etc.)	77) LOI Criteria
		9.2 SOI Reviews	78) SOI Review Strategy 79) Sampling Strategy
		9.3 Software Approval, including approval of Software Configuration Index (SCI) and Software Accomplishment Summary (SAS)	80) Software maturity evaluation for Type Inspection Authorization (TIA) 81) Open Problem Report (OPR) Evaluation 82) Software Change Impact Analysis (CIA) to determine Major or Minor Changes

Note: *In addition to the description of the items in the first three columns, the chapter number of the referenced DO-178C is also listed, such as 6.0 Software Verification, where 6.0 refers to DO-178C Chapter 6. The verification process is one of the 4 integral processes but is listed separately in the table because it is highly related to the software product. Each technology is identified as T_i, for instance, T_{81} refers to item 81) ORP technology in this table.*

There is a total of 82 technologies identified based on the DO-178C software life cycle. The same technology may be used in different processes, but the focus will be on different perspectives. For example, MBD may be used in planning, design, coding, and verification processes. In the planning phase, attention should be paid to the life cycle model, tools used, and modelling standards, but in the verification process, and the model review and model simulation should be concerned. Different DALs of software need to meet different DO-178C Objectives. Some technologies may apply to a higher level, but not be used for the lower level. The technology distribution statistics in each process are shown in Table 5 and *Figure 5*.

Table 5. Software Process Technology Coverage (PTC) matrix

DO-178C Process	Technology Coverage	Amount
Planning Process	T_1 ~T_{25}	25
Development Process	T_{26} ~T_{42}	17
Verification Process	T_{43}~ T_{67}	25
Configuration Process	T_{68}~ T_{74}	7
Quality Assurance Process	T_{75}~ T_{76}	2
Certification Liaison Process	T_{77} ~T_{82}	6

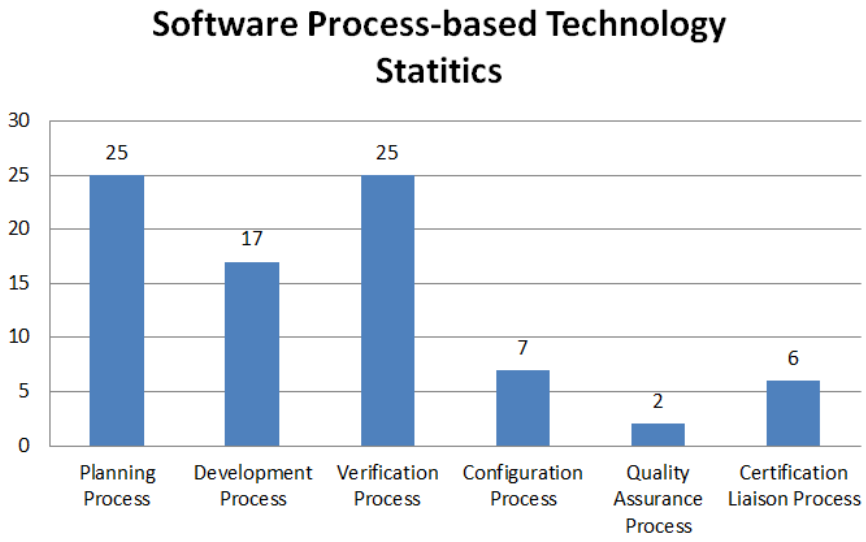

Figure 5. A quantitative analysis of the technology distribution of each process

5. ANALYSIS OF SOFTWARE REVIEWS AND SAMPLING STRATEGY

5.1. Analysis of SOI Reviews and LOI

According to FAA and EASA policy, four SOI reviews are defined, which are SOI#1 Software Planning Review, SOI#2 Software Development Review, SOI#3 Software Verification Review, and SOI#4 Final Certification Software Review (FAA, 2003) (EASA, 2012). The review aims to find systemic problems in the applicant's software developing processes and non-compliance issues with regulations and to establish confidence in the software through the reviews (FAA, 2004). The purpose of software SOI review is to ensure compliance with DO-178C objectives and other applicable software policy, guidance, and issue papers (FAA, 2018). The reviews can be conducted by a certification officer or delegated to a DER or ODA/DOA. The LOI depends on the project-specific conditions:

a) Combined reviews. If the software is level D, PDS with no change, minor changes of PDS, TSOA, very small size software project, implementing very simple system functions and so on, the software SOI reviews may be combined into less than four times (FAA, 2003) (EASA, 2012).
b) Increase review frequency. If it is a complex large project with many sub-systems, IMA architectures, Level A newly developed software, experiencing multiple problems during software testing or system testing, the new applicant with no DO-178B/C successful project experience, they can consider increasing the LOIs, especially in the development and verification process, which is recommended to do multiple SOI#2 and SOI#3 (FAA, 2003).
c) The main factors that can affect the SOI frequency are analysed as follows:

    1) the software category, which means PDS, COTS, new-developed software, TSOA software, libraries, RTOS, IMA hosted software, etc.,
    2) the software DALs as determined by the system safety assessment process (EASA, 2012) (FAA, 2003),
    3) the project characteristics, such as the tier of supplier-chain, the experience of the applicant, the complexity of the project, system functionality and novelty, software developing team human resources, and existence of issues associated with Section 12 of DO-178C (FAA, 2003),
    4) the use of new technologies or unusual design features (EASA, 2012),
    5) using alternative methods to show compliance,
    6) the establishment and operation of the software assurance aspect of the applicant's Design Assurance System (DAS), and
    7) the amount of planning review activities of the delegation systems (e.g. DER or ODA) and the applicant's self-monitoring status (EASA, 2012).

5.2. Analysis of SOI Review and Sampling Strategy

Studies indicate that developing a scientific and reasonable software review and sampling strategy, and mastering the technology related to each SOI review, especially the impact of this technology on software compliance verification, will facilitate the rapid identification of key clues during software reviews (Dodd & Habli, 2012). Each SOI review and sampling strategy, and the applicable identified technologies for each SOI are analysed in the following sections.

The goal of SOI#1 is to evaluate the compliance of the software planning with the applicable objectives of Table A-1 and A-8~A-10 of DO-178C Annex A (FAA, 2004).

The goal of SOI#2 is to assess whether the software plans and standards are effectively implemented and to evaluate the compliance of the software development process to the applicable objectives of DO-178C Table A-2~A-5, and A-8~A-10 (FAA, 2003). Reviewing is suggested that focus on the output of the software requirements process, design process, coding process, and integration process, and assess the compliance with applicable objectives of DO-178C Table A-2~A-5 through top-down and bottom-up thread review with the Risk-Based sampling strategy (VanderLeest, 2013)(Xing & Mu, 2015).

The purpose of SOI#3 is to evaluate the compliance of the software verification process with the applicable objectives of DO-178C Table A-6, A-7, and A-8~A-10 to assess the effectiveness and implementation of verification plans and procedures (FAA, 2003). The SOI#3 software review and sampling strategy are suggested to be also risk-based to evaluate the follow-up activities of findings, observations, and action items generated from the previous stage review.To perform a delta review of the development data if there are major changes from the previous review.To

assess the test cases, test procedures, verification results, test coverage, and code structure coverage to the applicable objectives of DO-178C Table A-6 and A-7. The sampling strategy is the same as SOI#2.

The goal of SOI#4 is to determine compliance of the final software product with the appropriate objectives of RTCA/DO-178C and other applicable certification policies and guidance (FAA, 2003). The SOI#4 review strategy is to evaluate the closure status of findings, observations, and action items of the previous reviews.Conduct a delta review of SOI#2 and SOI#3 when necessary if there are major changes or the reviewer does have sufficient confidence in the software product.Review the Software Conformity Review (SCR) record, or participate in the applicant's SCR meetings, which can be combined with the FAI to improve certification efficiency (Chen, et al., 2015).Assess the OPRs to judge whether it can be deferred to post-TC.Review the final SCI, SAS, tool qualification data, such as Tool Accomplishment Summary (TAS) if applicable, to ensure the version of software product intended to be used in the certified system or equipment fully comply with all applicable DO-178C objectives, the policy, and guidance (FAA, 2004).

5.3. Quantitative Analysis of SOI Technology & Objective Coverage

Through the above analysis, it can be known that the airborne software safety assurance can be achieved by a structured approach international best practice as described in DO-178C based on the process and objectives. Table 6 is the analysis result of the applicable technology and objectives of each SOI. The analysis approach and process are as follows:

a) Based on the analysis of the SOI review strategy in Section 4.3.2 of this paper, identify the applicable technologies associated with each SOI by referring to the technology list in **Table** .
b) Based on DO-178C Annex A and the analysis of SOI review strategy in Section 4.3.2 of this paper, in conjunction with FAA Order 8110.49 Chapter 2 "Software Review Process" (FAA 2003), which was based on DO-178B, analyzing these data to identify applicable objectives for each SOI based on DO-178C.

Figure *2* is the quantitative analysis of the distribution of TOC of each SOI review, which demonstrated that 50% of the DO-178C objectives are assessed in SOI#2 review, and 35% of technologies are related and are among the highest proportions. According to the number of objectives, SOI#3 is the second, with the objectives accounting for 32%, and the technology points involved accounting for 26%. Only SOI#1's technology accounting for 31%, but SOI#1's objective accounting for 16%, which is the third. Finally, SOI#4 objectives are the least 2%, and technology accounts for 8%. However, SOI#4 is a review of the entire life cycle process. It is necessary to evaluate all previous SOI review opening items, non-conformance items, and observation items. Therefore, although the SOI#4 objectives are accounted for the least, it plays a very critical role in the entire software review process, because through SOI#4 the reviewers will determine whether the software is in compliance with all the applicable objectives of DO-178C and whether it can obtain the final approval.

Table 6. TOC Matrix of each SOI.
Source: (FAA, 2004) (RTCA, 2011a)

SOI	Technology		Objectives	
	Identification	**Amount**	**Identification**	**Amount**
SOI#1	T_1~T_{25}, T_{43}, T_{68} ~ T_{69}, T_{77}~T_{78}	30	Table A-1: Objective1-7(All Objectives) Table A-8: Objective1-4 Table A-9: Objective 1 Table A-10: Objective1-2	14
SOI#2	T_{26} ~T_{42} T_{44}~ T_{54} T_{68}~T_{71} T_{78}~T_{79}	34	Table A-2: Objective 1-6 Table A-3: Objective1-7(All Objectives) Table A-4: Objective 1-13(All Objectives) Table A-5: Objective 1-6 Table A-8: Objective 1-4,6 Table A-9: Objective 1-4 Table A-10: Objective 1-2	43
SOI#3	T_{48}~T_{49}, T_{51}, T_{54} T_{55} ~ T_{67} T_{68}~ T_{71} T_{77}~T_{81}	26	Table A-5: Objective 7-9 Table A-6: Objective1-5(All Objectives) Table A-7: Objective 1-9(All Objectives) Table A-8: Objective1-6(All Objectives) Table A-9: Objective1-4 Table A-10: Objective 1-2	28
SOI#4	T_{75} ~ T_{82}	8	Table A-9: Objective 5 Table A-10: Objective 3	2

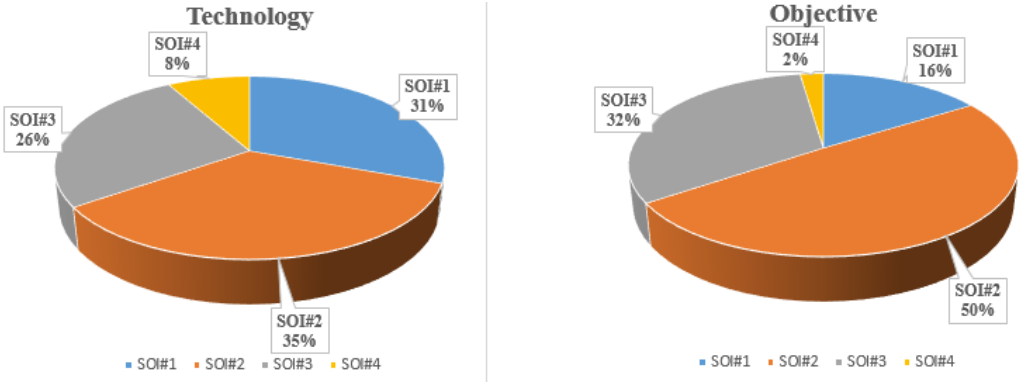

Figure 6. The TOC distribution of each SOI

6. CONCLUSIONS

Software reviews are always treated as a critical part of the system certification process, provided that it is conducted in accordance with each NAA's procedures and handbooks to finding compliance with the safety-related regulations § 25.1301 and § 25.1309. This research analysed regulation requirements and software review policies of the FAA, EASA, CASA, and CAAC using a comparative approach to establish the software certification basis and compliance means. Given that the airborne software review is performed by people, the different working experiences, backgrounds, and technical capabilities of the reviewers may lead to different review conclusions.

An in-depth software review can discover the holes existing in the design and potential risks of the aircraft. This paper studied the technical focuses of airborne software review based on the DO-178C software life cycle process and identified 82 technology aspects through analysis of objectives and activities of each process. This paper also analysed the LOI impact factors of airborne software SOI review and developed a set of Risk-based SOI reviews and sampling strategy taking into account the applicable identified technologies and compliance objectives of DO-178C by developing the PTC and TOC matrixes. The study of this paper will help NAAs to maintain software expertise and formulate more effective software review procedures and guidance documents, and carry out corresponding technical research to ensure aircraft safety by conducting in-depth software reviews from a software certification perspective.

Due to the time constraints, the limitation of this study is that the analysis of the technology focused on the software life cycle process and the research of software review and sampling strategy in this research is based on Level A software, which is also the most severe safety level, however, this study did not distinguish the application scope of different DALs.
In the research process of this project, it was found that an Objective-oriented SOI review method based on DO-178C is meaningful. On the one hand, it is helpful for the software reviewers to judge the compliance, on the other hand, it can provide effective assistance for the applicant to demonstrate compliant evidence and perform software verification activities. Therefore it demonstrated the necessity of a future study to explore the applicable technical focuses and SOI review strategies for different DALs of airborne software based on each objective of DO-178C.

REFERENCES

[1] Anderson, L. et al., 2014. DO-178B/C Differences Tool. [Online] Available at: https://www.faa.gov/aircraft/air_cert/design_approvals/air_software/media/differences_tool.pdf [Accessed 7 Aug. 2020].

[2] Beautiful, I. i., 2015. Codebases. [Online] Available at: https://informationisbeautiful.net/visualizations/million-lines-of-code/ [Accessed 8 Aug. 2020].

[3] Berk, R. E., 2009. An Analysis of Current Guidance in the Certification of Airborne Software, Massachusetts: MASSACHUSETTS INSTITUTE OF TECHNOLOGY.

[4] Boeing, 2020. 737Max Software Update. [Online] Available at: https://www.boeing.com/commercial/737max/737-max-software-updates.page [Accessed 25 10 2020].

[5] CAAC, 2000. AC-21-02, Beijing: CAAC.

[6] Cai, Y., 2020. CAAC SOI Review Policy [Interview] (20 Oct. 2020).

[7] Cai, Y. et al., 2013. Research on Airborne Software Airworthiness Standards DO-178B/C. 1st ed. Shanghai: Shanghai Jiaotong University Press.

[8] CASA, 2011. Industry Delegates Management, Canberra: CASA.

[9] CASA, 2014. AC 21-50: Approval of software and electronic hardware parts, Canberra: CASA.

[10] CASA, 2019. Approved Design Organisations (Subpart 21.J). [Online] Available at: https://www.casa.gov.au/licences-and-certification/aircraft-certification-and-design/approved-design-organisations-subpart-21j [Accessed 19 Oct. 2020].

[11] CAST, 2002. CAST-10 "Literal" Interpretation of Decision Coverage Increases Rigor of Testing Requirements. [Online] Available at: https://www.rapitasystems.com/blog/cast-10-literal-interpretation-decision-coverage-increases-rigor-testing-requirements [Accessed 6 July 2020].

[12] Chen, Y., Yan, L. & Sun, J., 2015. Civil Aircraft Airborne Software Management. 1st 编辑 Beijing: The Aviation Industry Press of China.

[13] COMMITTEE, T. H., 2020. FINAL COMMITTEE REPORT: THE DESIGN, DEVELOPMENT & CERTIFICATION OF THE BOEING 737 MAX, USA: THE House COMMITTEE on TRANSPORTATION AND INFRASTRUCTURE.

[14] Delange, J. et al., 2015. Evaluating and Mitigating the Impact of Complexity in Software Models, Pittsburgh: Carnegie Mellon University.

[15] Dewalt, M. P., 1988. Comparison of FAA DO-178A and DOD-STD-2167A approaches to software

certification. San Jose, 17 October 1988 - 20 October 1988 Digital Avionics Systems Conference.
[16] Dodd, I. & Habli, I., 2012. Safety certification of airborne software: An empirical study. Reliability Engineering & System Safety, 98(1), pp. 7-23.
[17] EASA, 2012. EASA CM – SWCEH – 002 Issue: 01 Revision: 01 Software Aspects of Certification. [Online] Available at: https://www.easa.europa.eu/sites/default/files/dfu/certification-docs-certification-memorandum-EASA-CM-SWCEH-002-Issue-01-Rev-01-Software-Aspects-of-Certification.pdf [Accessed 19 Oct. 2020].
[18] EASA, 2013. AMC 20-115C: Software Considerations for Certification of Airborne Systems and Equipment, Cologne: EASA.
[19] EASA, 2017. AMC 20-115D: AIRBORNE SOFTWARE DEVELOPMENT ASSURANCE USING EUROCAE ED-12 AND RTCA DO-178, Cologne: EASA.
[20] FAA, 1982. AC 20-115: Airborne Software Development Assurance Using EUROCAE ED-12 and RTCA DO-178, Washington: FAA.
[21] FAA, 1988. AC 25.1309-1A: SYSTEM DESIGN AND ANALYSIS, Washington: FAA.
[22] FAA, 2003. FAA Order8110.49: SOFTWARE APPROVAL GUIDELINES. [Online] Available at: https://www.faa.gov/documentLibrary/media/Order/FAA_Order_8110.49.pdf [Accessed 19 Oct. 2020].
[23] FAA, 2004. Job Aid: Conducting Software Reviews Prior to Certification. [Online] Available at: https://elsmar.com/elsmarqualityforum/attachments/jobaid-r1-1-pdf.14401/ [Accessed 19 Oct. 2020].
[24] FAA, 2007. 14 CFR § 25.1301 - Function and installation. [Online] Available at: https://www.law.cornell.edu/cfr/text/14/25.1301 [Accessed 3 Oct. 2020].
[25] FAA, 2007. 14 CFR § 25.1309 - Equipment, systems, and installations.. [Online] Available at: https://www.law.cornell.edu/cfr/text/14/25.1309 [Accessed 19 Sep. 2020].
[26] FAA, 2008. 14 CFR § 33.28 - Engine control systems. [Online] Available at: https://www.law.cornell.edu/cfr/text/14/33.28 [Accessed 28 Aug. 2020].
[27] FAA, 2011. Order8110.49 Chg1: Software Approval Guidelines, Washington: FAA.
[28] FAA, 2013. AC20-115C: Airborne Software Assurance, Washington: FAA.
[29] FAA, 2017. AC 20-115D: Airborne Software Development Assurance Using EUROCAE ED-12( ) and RTCA DO-178( ), Washington: FAA.
[30] FAA, 2017. Airborne Software Development Assurance Using EUROCAE ED-12( ) and RTCA DO-178( ). [Online] Available at: https://www.faa.gov/regulations_policies/advisory_circulars/index.cfm/go/document.information/documentID/1032046 [Accessed 19 10 2020].
[31] FAA, 2018. Order 8110.49 A: Software Approval Guidelines. [Online] Available at: https://www.faa.gov/documentLibrary/media/Order/FAA_Order_8110.49A.pdf [Accessed 19 Oct. 2020].
[32] FAA, 2018. Order8110.49 A: Software Approval Guidelines, Washington: FAA.
[33] FAA, 2020. Certification Authorities Software Team (CAST). [Online] Available at: https://www.faa.gov/aircraft/air_cert/design_approvals/air_software/cast/ [Accessed 10 9 2020].
[34] Hayhurst, K. J., Veerhusen, D. S., Chilenski, J. J. & Rierson, L. K., 2001. A Practical Tutorial on Modified Condition/Decision Coverage, Hanover: NASA.
[35] Hilderman, V. & Baghai, T., 2007. Avionics Certification: A Complete Guide to DO-178 (Software), DO-254 (Hardware). 1st ed. Covina: Avionics Communications, Incorporated.
[36] Hughes, W. J., 2016. Software Assurance Approaches, Considerations, and Limitations: Final Report, Washington: FAA.
[37] ICAO, 2005. Annex 8: Airworthiness of Aircraft. 10th ed. s.l.: ICAO.
[38] Jimenez, J. A. et al., 2020. A Framework for Evaluating the Standards for the Production of Airborne and Ground Traffic Management Software. IEEE Access, 8(1), pp. 142-161.
[39] LU, Y. et al., 2011. Coverage analysis of airborne software testing based on DO178B standard. Procedia Engineering, I(17), pp. 480-488.
[40] Marques, J. & Yelisetty, S., 2019. AN ANALYSIS OF SOFTWARE REQUIREMENTS SPECIFICATION CHARACTERISTICS IN REGULATED ENVIRONMENTS. International Journal of Software Engineering & Applications (IJSEA), 10(6), pp. 1-15.
[41] McCormick, G., 2015. Certification of Civil Avionics. In: G. R. Spitzer, U. Ferrell & T. Ferrell, eds. The Third Edition： Digital Avionics Handbook. Boca Raton: CRC Press LLC, p. Chapter 23.
[42] McCormick, G., Bryan, M. & DeWalt, P., 2009. Confusions of FAA Software Review Job Aid [Interview] (12 3 2009).

[43] Mendis, K. S., 2008. Software Safety and Its Relation to Software Quality Assurance. In: G. G. Schulmeyer, ed. Handbook of Software Quality Assurance. Boston: ATECH HOUSE, p. 211.
[44] Rash, W., 2019. eWEEK. [Online] Available at: https://www.eweek.com/mobile/how-software-can-make-an-airplane-crash [Accessed 30 July 2020].
[45] Rierson, . L., 2013. Developing Safety-Critical Software: A Practical Guide for Aviation Software and DO-178C Compliance. 1 ed. Boca Raton: CRC Press.
[46] Romanski, G., 2001. The Challenges of Software Certification. [Online] Available at: https://www.researchgate.net/publication/228778585_The_Challenges_of_Software_Certification [Accessed 19 10 2020].
[47] RTCA, 1982. DO-178: Software Considerations in Airborne Systems and Equipment Certification, Washington: RTCA.
[48] RTCA, 1985. DO-178A: Software Considerations in Airborne Systems and Equipment Certification, Washington: RTCA.
[49] RTCA, 1992. DO-178B: Software Considerations in Airborne Systems and Equipment Certification, Washington: RTCA.
[50] RTCA, 2011a. DO-178C: Software Considerations in Airborne Systems and Equipment Certification, Washington: RTCA, Inc.
[51] RTCA, 2011b. DO-248C: Supporting Information for DO-178C and DO-278A, Washington: RTCA, Inc.
[52] Squair, M. J., 2006. Issues in the Application of Software Safety Standards, Sydney: Australian Computer Society, Inc.
[53] StClair, B. & Hillary, N., 2012. DO-178C: Improved certification for cost-effective avionics systems, USA: LDRA.
[54] Sun, A., Cheng, W. & Wang, M., 2019. In: 1st, ed. System Safety. Beijing: China Civil Aviation Publishing House, pp. 111-233.
[55] VanderLeest, S. H., 2013. Wikipedia. [Online] Available at: https://en.wikipedia.org/wiki/DO-178C#/media/File:DO-178C_Traceability.png [Accessed 19 Oct. 2020].
[56] Wetherholt, M. J. & Penix, J., 2002. System Software Safety: Today's Practical Approach versus Tomorrow's Promise. Noordwijk, ESA-NASA.
[57] Xing, L. & Mu, M., 2015. Research On Airworthiness Standard DO-178B / C′s Object Analysis and Stage of Involvement Review in Airborne Software. Aeronautical Computing Technique, 45(5), pp. 97-101.
[58] Yanyun, W., 2016. Developing Safety-Critical Embedded Software under DO-178C. [Online] Available at: https://etd.ohiolink.edu/!etd.send_file?accession=ucin1468575365&disposition=inline [Accessed 12 Sep. 2020].
[59] Yelisetty, S., Marques, J. & Tasinaffo, P. M., 2015. A set of metrics to assess and monitor compliance with RTCA DO-178C. 2015 IEEE/AIAA 34th Digital Avionics Systems Conference (DASC), 10.1109(DASC.2015.7311644), pp. 1-18.