



Civil Aviation Administrator of China

Advisory Circular

No. : AC-21-02

Date : 10/JAN/2000

Software Review Methods for Airborne System and Equipment Certification

Airplane Airworthiness Certification Department of CAAC

Aircraft Airworthiness Certification Department of CAAC

Advisory Circular

No. : AC-21-02

Initiated by : CDC

Date : 10/JAN/2000

Approved by : Wang Zhong

**Software Review Methods for Airborne System and
Equipment Certification**

Contents

1. Purpose
2. Basis and Backgrounds
3. Related documents
4. DO-178B/DE-12B and software review
5. Software level and failure condition categorization
6. Overview of the software review
 - 6.1 Software review and software life cycle
 - 6.2 Basic processes and methods of software review
 - 6.3 Clarifications of alternative methods
 - 6.4 Requirements for adopting previously developed software
 - 6.5 Software develop and verification tools qualification
7. Basic responsibilities and tasks of designated engineering representative(DER)
8. Software quality assurance
9. Annex(process objectives and outputs by software level)

Software review methods for airborne system and equipment certification

1. purpose

This advisory circular briefly explains how to conduct airborne software airworthiness review activities using RTCA/DO-178B or EUROCAE/ED-12B “software considerations of airborne software and equipment certification” during airborne software or equipment certification process.

This advisory circular aims to provide guidance for airworthiness certification authorities and applicants and its suppliers to understand and master requirements of RTCA/DO-178B and EUROCAE/ED-12B (hereafter referred to as DO-178B and ED-12B) better, and then conduct airborne software airworthiness compliance verification and certification activities with higher level of confidence.

As a guidance document, methods provided by this advisory circular are either exclusive or compulsory.

2. basis and backgrounds

This advisory circular is formulated on the basis of domestic airborne software review practice, in accordance with DO-178B/ED-12B with referring to Boeing tutorial material “Function of Certification

Engineer in Airborne System and Equipment Certification”. With accumulation of software review experiences, this circular will be revised and improved constantly from now on.

3. related documents

CCAR-parts 21, parts 23, parts 25, parts 27, parts 29, parts 183;

RTCA/DO-178B, EUROCAE/ED-12B;

AC 25.1309-1A/AMJ 25-1309, AC 23.1309-1C ;

4. DO-178B/ED-12B and software review

DO-178B/ED-12B provides a series of means with an acceptable level of confidence for current airborne software certification process with respect to how to ensure the software aspects of airborne system and equipment comply with airworthiness requirements. It applies to applicant and its suppliers of type certificate(TC)/type accreditation certificate(VTC)、supplementary type certificate(STC)/ supplementary type accreditation certificate(SVTC) or TSOA/PMA/VDA. Figure 4-1 illustrates practice aspects of DO-178B/ED-12B in airworthiness certification process.

Airworthiness authority

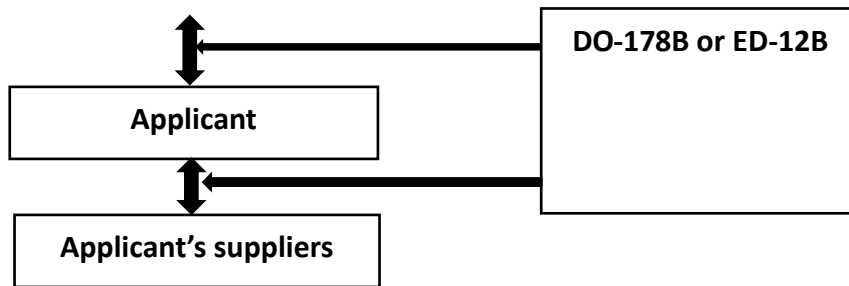


Figure4-1 practice aspects of DO-178B/ED-12B

DO-178B/ED-12B was jointly approved and released by RTCA (radio technical commission for aeronautics) and EUROCAE (European organization for civil aviation equipment) at December 6th, 1992 on the basis of RTCA/DO-178A (or EUROCAE/ED-12A) and in accordance with new practical experiences of airborne software review after 1985.

DO-178B/ED-12B provides guidance listed below on how to comply with airworthiness requirements:

- ☆ assure the corresponding relationship between software level and objectives for software life cycle processes;

- ☆ assure objectives for software life cycle processes;

- ☆ explain means for satisfying those objectives and design considerations;

- ☆ explain how to indicate that the objectives have been satisfied.

Table A-1 to table A-10 of DO-178B/ED-12B annex A (process objectives and outputs by software level) express those guidance in summary. For details of those tables, please refer to annex of this

document.

5. software level and failure condition categorization

Software level is determined by its failure condition category. The category definition of software failure condition is exactly the same with that of system failure condition. Meanwhile, software failure condition category and system failure condition category are both determined by system safety assessment process, i.e. determined while determining certification basis of system. However, differs from hardware, developing software by a software level doesn't mean failure rate has been allocated to that, therefore during the system safety assessment process, software level or software reliability rate that based on a specified software level can't be allocated like hardware failure rate. However, it has no effect on the fact that software modules (components) are connected in series or parallel in the multi-module (component) partitioned software. So that the failure of software module (component) of that would result in different influence on a certain software function.

For definition of failure condition categorization and software level as well as guidance for determining software level, please refer to section 2.2 of DO-178B/ED-12B. Figure 5-1 presents corresponding relationship between software level and failure condition categorization with brief description for certain software level.

Failure condition	Software level	Brief description
catastrophic	A	Failure condition will result in: the airplane's incapability of normal flight and landing.
hazardous	B	Failure condition will result in: severely reduction of capability of the airplane or ability of the flight crew to cope with adverse operating conditions.
Major	C	Failure condition will result in: dramatically reduction of capability of the airplane or ability of the flight crew to cope with adverse operating conditions.
Minor	D	Failure condition will result in: slightly reduction of capability of the airplane or ability of the flight crew to cope with adverse operating conditions.
No safety effect	E	Failure condition will result in: no effect on any capability of the airplane

		or any ability of flight crew.
--	--	--------------------------------

Figure 5-1 relationship between software level and failure condition category

6. overview of the software review

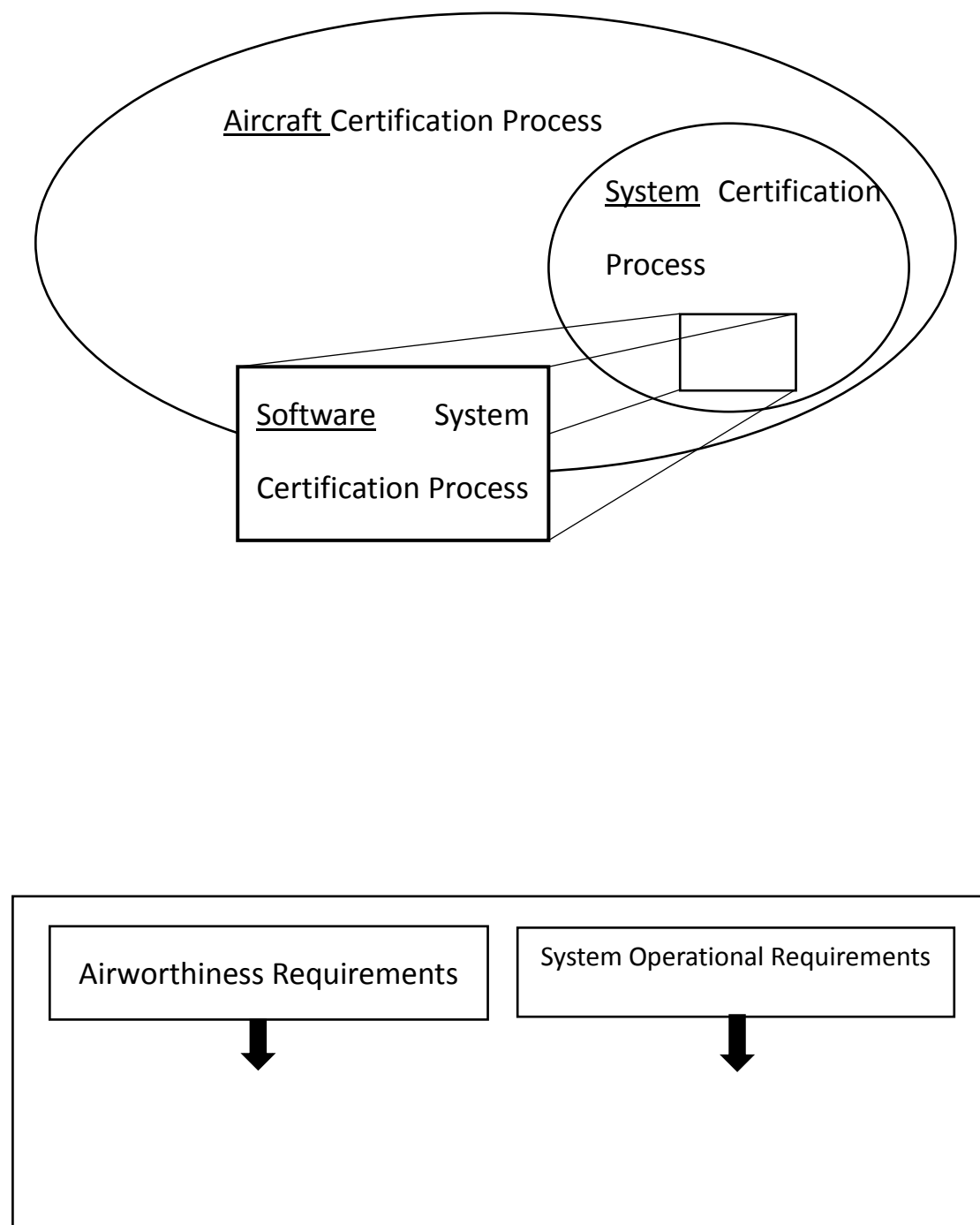
At the stage of confirming certification basis, through processes of system requirements distribution and system safety assessment, software functions and software level allocated by system are determined. Thereby comes the stage of software development and review. Following items briefly explain every software life cycle processes and relationship between each processes as well as basic software review procedures and methods during the whole stage of software development and review.

6.1 software review and software life cycle processes

Software review is an indispensable step to achieve type/system certification goal. As Figure 6-1 shows, software review comes from software type/system review process, and eventually returns and ends in software type/system review process. Software life cycle processes include: software planning process, software development processes and software integral processes that runs through those former processes. Meanwhile, software development processes are further subdivided into software requirement process, software design process, software coding

process and software integration process----all of them compose the main part of software product development. And software integral processes are further subdivided into software verification process, software configuration management process, software quality assurance process and certification liaison process. Among them software verification process composes the main part of software review.

The basic inter-relationship between those processes is presents by Figure 6-2



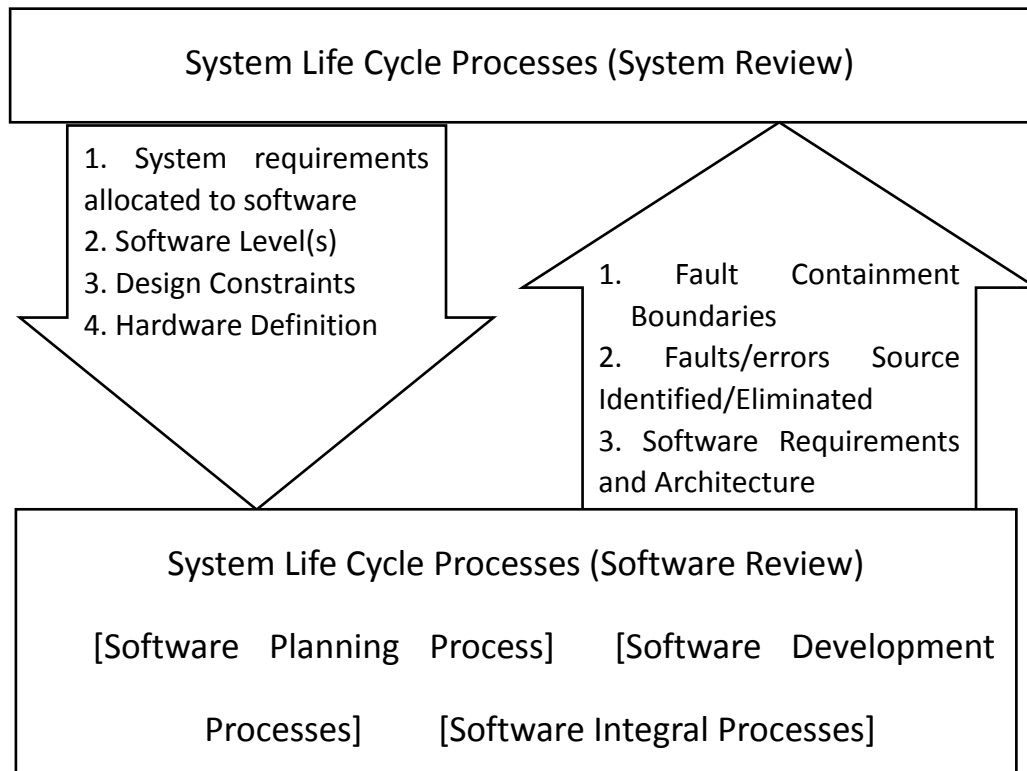
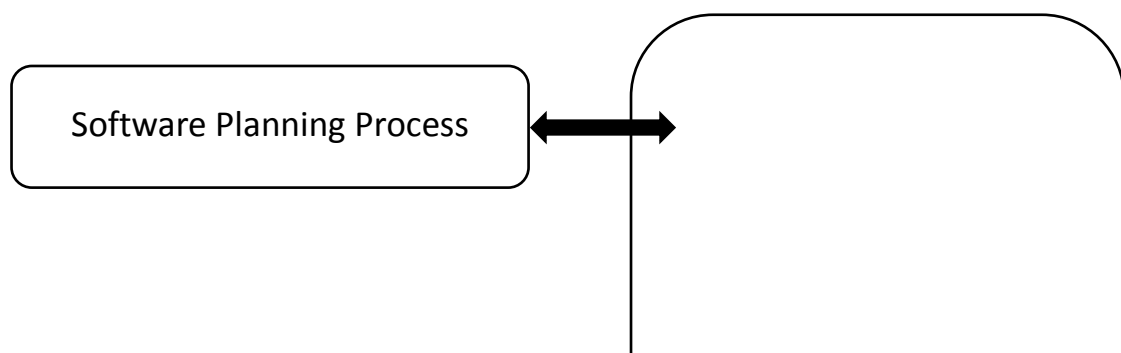


Figure 6-1 relationship between software review processes and Type/Software airworthiness certification processes.



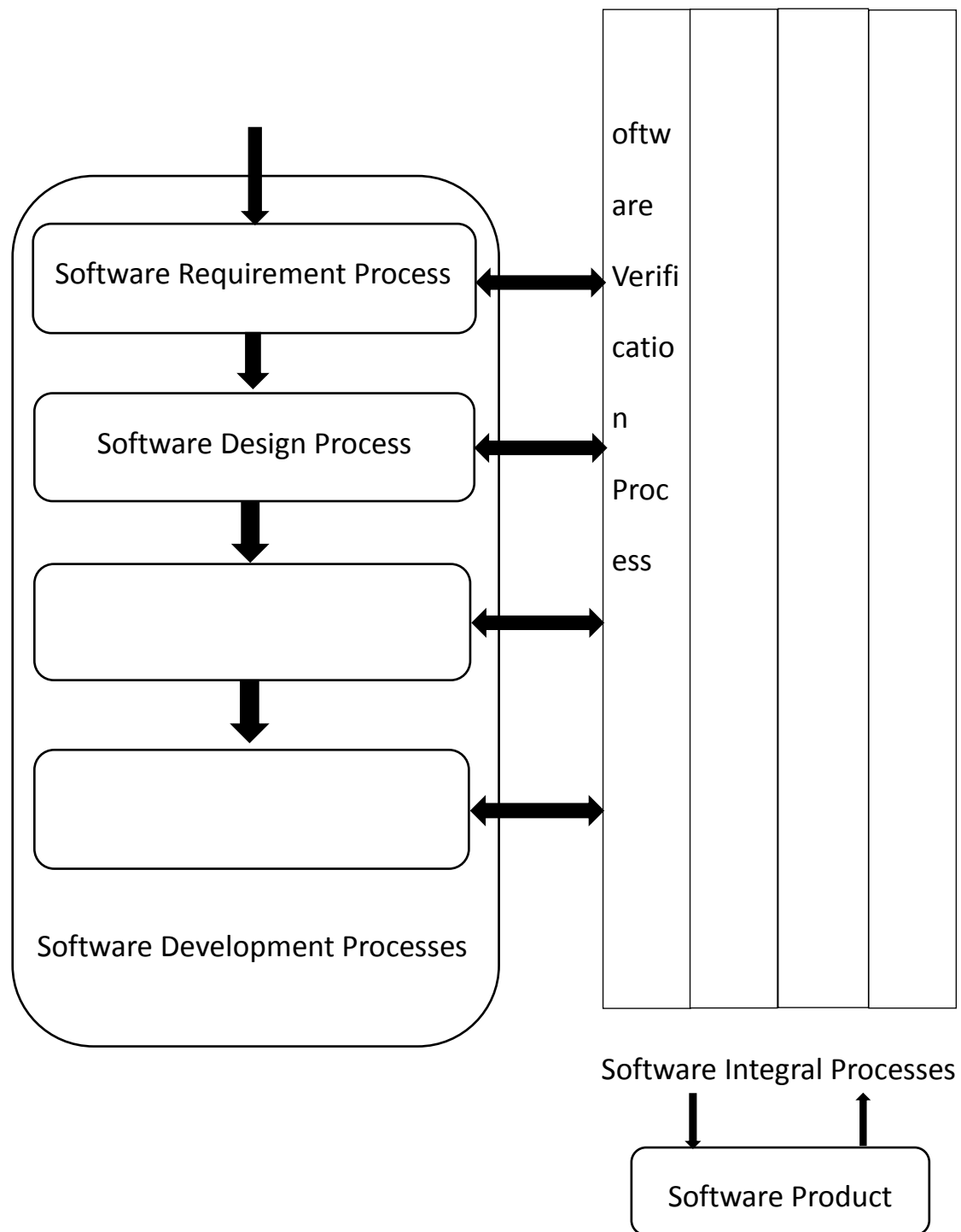


Figure 6-2 basic relationships between software life cycle processes

6.2 basic processes and methods of software review

The particularity of software life cycle processes determines the characteristic of software review methods during processes to ensure safety (reliability), traceability, verifiability and maintainability of

software product. Figure 6-3 shows the basic procedure and method of software review, which determined by software level that determined in Type/System certification basis. Firstly, determine software's objectives and means of compliance in every software life cycle processes according to DO-178B/ED-12B annex A (Lifecycle objectives and outputs by software level) table A-1~A-10 with respect to certain software level. Then review applicant's compliance outputs for those objectives item by item. Among them plan for software aspects of certification, software configuration index and software accomplishment summary are three basic documents for airworthiness authority to review and approve applicant's means of compliance and evidence of compliance. It's necessary to pay attention that the means of compliance and airworthiness objectives offered by DO-178B/ED-12B are of principle, applicant should make those airworthiness objectives and means of compliance specific by applying professional knowledge of software engineering based on the realistic situation of target software product. In order to indicate compliance to software requirements of certification basis, applicant should organize "software compliance checklist" referring to table A-1~A-10 and submit to certification engineer. Firstly, certification engineer should focus on reviewing plan for software aspects of certification and approve means of compliance adopted by applicant. Then review the compliance situation of software

development processes and integral processes. The key point of which is reviewing and analyzing the traceability and verifiability of software and checking up regulations(procedure) and result of verification test (including trial flight if necessary) termwise as well as coverage analysis for software verification result, so as to ensure that the applicant:

1. have implemented safety considerations for software;
2. Systems/equipment that should upload this software will compliance the safety requirements;
3. has the ability to produce approved software product continually;

Notes: if the applicant get accustomed to indicate the compliance to DO-178/ED-12B according to “stage” mode such as “requirement review stage(RR)”, “ preliminary design review stage(PDR)”, “critical design review stage(CDR)”, “test ready review stage(TRR)” and “first piece inspection stage(FAI)”, then applicant has to define corresponding relationship between those stages and software life cycle processes in plan for software aspects of certification and ensure that the software comply with relative objectives.

6.3 Clarifications of alternative methods

In the current context, the alternative methods refer to methods that comply with one or some objectives of DO-178B/ED-12B. Alternative methods can't be separated from the principle line, which is

software development processes. It's necessary to clarify that besides means of compliance provided by itself, DO-178B/ED-12B doesn't prohibit other means of compliance to satisfy airworthiness compliance requirements. If adopts alternative methods, the applicant shall:

1. indicates that the alternative methods will comply with one or some objective requirements of DO-178B/ED-12B;

2. indicates influences to software development processes and software life cycle documents introduced by the alternative methods and clarifies the rationality of adopting the alternative methods(id est. indicates that adopting the alternative methods still satisfy safety objective requirements of system) in plan for software aspects of certification;

3. acquires approval from airworthiness authorities;

At present, common alternative methods include: formal method for software development processes, exhaustive input testing, verification method for multi-version dissimilar software, software reliability model that could be used in software verification processes and product operational experiences. Section 12.3 of DO-178B/ED-12B provides guidance for those alternative methods.

For early software approved by DO-178A/ED-12A, DO-178B/ED-12B also can be considered as alternative methods. It's necessary to pay attention that: since DO-178A/ED-12A didn't provide enough means of

compliance for user-modifiable software (such as database), option-selectable software, software development and verification tools, previously developed modular software and field loadable software, it's necessary to indicate compliance to airworthiness requirements of DO-178B/ED-12B if those early software shall be used.

6.4 Requirements for adopting previously developed software(including software modification situations)

In the current context, previously developed software refers to the software that is developed according to life cycle processes or equivalent processes of DO-178B/ED-12B.

One or some of situations listed below will be involved when using this category of software:

1. modifying previously developed software due to type/system requirements or operational experiences;
2. applying approved software of some type of aircraft to another new type of aircraft;
3. installing this category of software on new processor or hardware, or integrate that with other new software or modify software development environment;
4. upgrading development baselines(such as improving software level) of previously developed software due to aircraft

modification requirements;

Software configuration process and software quality assurance process should consider the reuse of previously developed software. Section 12.1 of DO-178B/ED-12B provides guidance for situations above.

6.5 software develop and verification tools qualification

Software tools could be classified into two categories, one is development tools and the other is verification tools.

Qualification of tools should be conducted when methods and processes of DO-178B/ED-12B are eliminated, reduced, or automated by the use of software tools without its output being verified as specified in section 6 of DO-178B/ED-12B.

The tool is qualified via tools qualification process. The purpose of tools qualification process is to ensure that the tools provides confidence at least equivalent to that of the processes are eliminated, reduced, or automated in DO-178B/ED-12B.

Section 12.2 of DO-178B/ED-12B provides guidance for tools qualification objectives, regulations, documents and its approval.

7. basic responsibilities and tasks of designated engineering representative

During the software review process, the task of software designated engineering representative (software DER for short) is to determine

whether the software satisfy the requirements stated in the certification basis by assessing software life cycle process and its compliance documents. Detail contents include:

1. assessing software plan and software product's compliance to specific standards and regulations;
2. assessing the implementation performance of those plans and procedures in development processes, verification process, configuration process and quality assurance process;
3. assisting DMIR to ensure that quality assurance system conducts supervision and review on software processes;
4. supervising and urging applicant to solve compliance problems exposed in software processes and software product processes;
5. providing compliance evidences of software to airworthiness authority;

Software DER's basic responsibilities include:

1. supervising software processes and its outputs' compliance to DO-178B/ED-12B;
2. be participated in assessing software problems with respect to safety;
3. be participated in preparing "software accomplishment summary" document;
4. be participated in coordinating certificate liaison process with

airworthiness authority;

5. informing applicant's suppliers with software problems that airworthiness authorities concern;

6. preparing supplemental materials about software review(such as job record of DER, review record, and test observation record etc.) and submit to airworthiness authority;

8. software quality assurance

In consideration of characteristic of software product, DO-178B/ED-12B particularly emphasize on quality assurance requirements of software design stage and design change stage.

During the software quality assurance process, applicant's quality assurance system should supervises and reviews the software life cycle processes and their outputs according to approved software quality assurance plan to obtain assurance that: deficiencies in any processes are detected, evaluated, tracked and resolved and software product and software life cycle data conform to certification requirements. The objectives of software quality assurance process are to obtain assurance that:

1. Software development process and integral process (including software modification) comply with approved software plan, specific standards and regulations via institutional framework and process

management procedure;

2. The transition criteria is satisfied when entering and exiting each software life cycle process;

3. A conformity review of the software product is conducted;

4. Approved software product are produced with consistency.

The same as other quality assurance activities, software quality assurance activities should have independence and authority, and should exert initiatives in other software life cycle process activities. For detail of software quality assurance process, please refer to section 8.2 and 8.3 of DO-178B/ED-12-B.

Determine software level (certificate basis)

DO-178B section 2.2

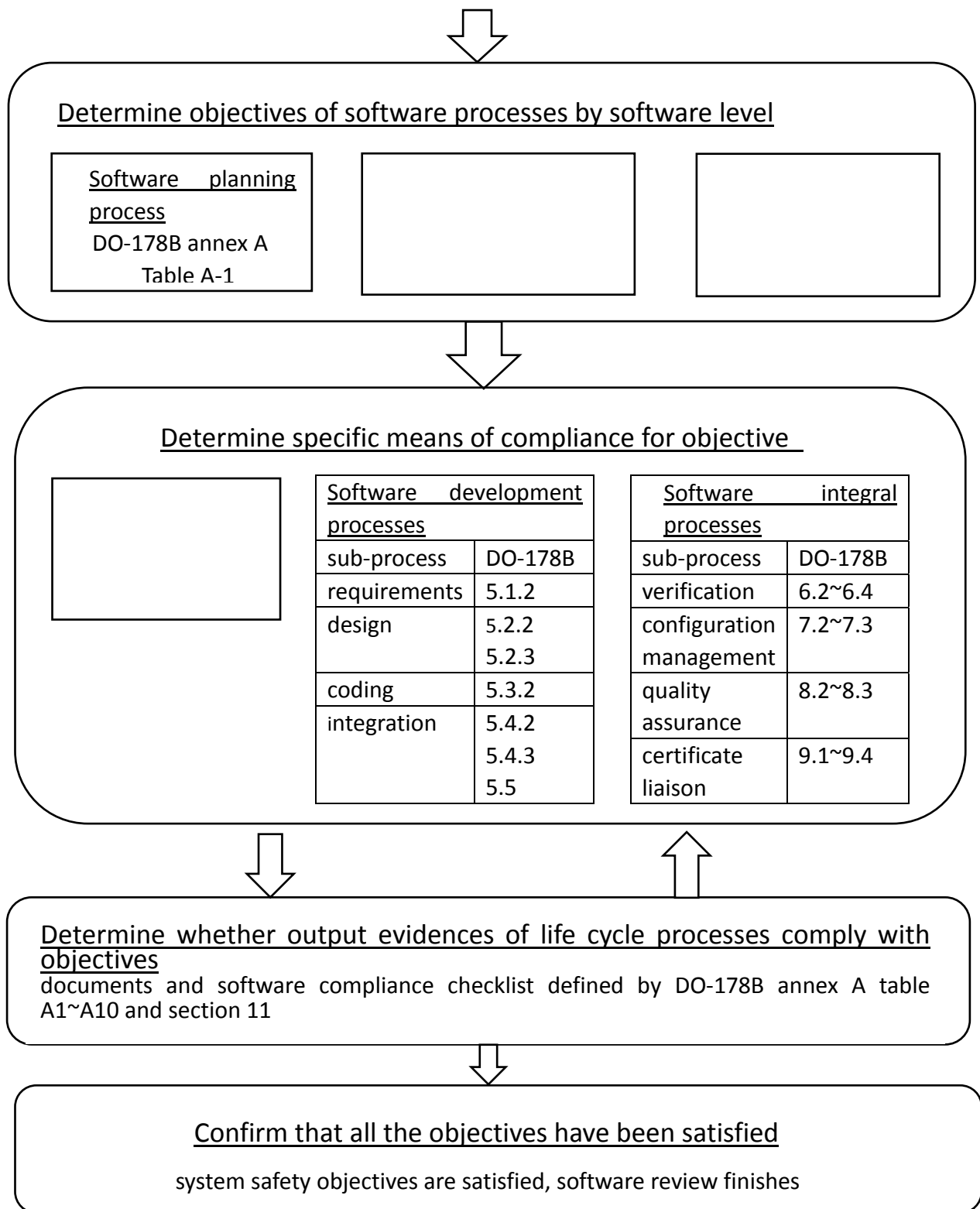


Figure 6-3 basic procedure and methods of software review

9. Annex 1 DO-178B/ED-12B annex A table A1~A10

Table A-1
Software Planning Process

Objective		Ref.	Applicability by SW Level				Output		Ref.	Control Category by SW level			
Description			A	B	C	D	Description			A	B	C	D
1 Software development and integral processes activities are defined.	4.1a 4.3		○	○	○	○	Plan for Software Aspects of Certification	11.1		①	①	①	①
							Software Development Plan	11.2		①	①	②	②
							Software Verification Plan	11.3		①	①	②	②
							SCM Plan	11.4		①	①	②	②
							SQA Plan	11.5		①	①	②	②
2 Transition criteria, inter-relationships and sequencing among processes are defined.	4.1b 4.3		○	○	○								
3 Software life cycle environment is defined.	4.1c		○	○	○								
4 Additional considerations are addressed.	4.1d		○	○	○	○							
5 Software development standards are defined.	4.1e		○	○	○		SW Requirements Standards	11.6		①	①	②	
							SW Design Standards	11.7		①	①	②	
							SW Code Standards	11.8		①	①	②	
6 Software plans comply with this document.	4.1f 4.6		○	○	○		SQA Records	11.19		②	②	②	
							Software Verification Results	11.14		②	②	②	
7 Software plans are coordinated.	4.1g 4.6		○	○	○		SQA Records	11.19		②	②	②	
							Software Verification Results	11.14		②	②	②	

LEGEND:

- The objective should be satisfied with independence.
- The objective should be satisfied.
- Blank Satisfaction of objective is at applicant's discretion.
- ① Data satisfies the objectives of Control Category 1 (CC1).
- ② Data satisfies the objectives of Control Category 2 (CC2).

Table A-2
Software Development Processes

Objective		Applicability by SW Level				Output		Control Category by SW level				
	Description	Ref.	A	B	C	D	Description	Ref.	A	B	C	D
1	High-level requirements are developed.	5.1.1a	○	○	○	○	Software Requirements Data	11.9	①	①	①	①
2	Derived high-level requirements are defined.	5.1.1b	○	○	○	○	Software Requirements Data	11.9	①	①	①	①
3	Software architecture is developed.	5.2.1a	○	○	○	○	Design Description	11.10	①	①	②	②
4	Low-level requirements are developed.	5.2.1a	○	○	○	○	Design Description	11.10	①	①	②	②
5	Derived low-level requirements are defined.	5.2.1b	○	○	○	○	Design Description	11.10	①	①	②	②
6	Source Code is developed.	5.3.1a	○	○	○	○	Source Code	11.11	①	①	①	①
7	Executable Object Code is produced and integrated in the target computer.	5.4.1a	○	○	○	○	Executable Object Code	11.12	①	①	①	①

LEGEND:

- The objective should be satisfied with independence.
- The objective should be satisfied.
- Blank Satisfaction of objective is at applicant's discretion.
- ① Data satisfies the objectives of Control Category 1 (CC1).
- ② Data satisfies the objectives of Control Category 2 (CC2).

Table A-3
Verification of Outputs of Software Requirements Process

Objective			Applicability by SW Level				Output		Control Category by SW level			
	Description	Ref.	A	B	C	D	Description	Ref.	A	B	C	D
1	Software high-level requirements comply with system requirements.	6.3.1a	●	●	○	○	Software Verification Results	11.14	②	②	②	②
2	High-level requirements are accurate and consistent.	6.3.1b	●	●	○	○	Software Verification Results	11.14	②	②	②	②
3	High-level requirements are compatible with target computer.	6.3.1c	○	○			Software Verification Results	11.14	②	②		
4	High-level requirements are verifiable.	6.3.1d	○	○	○		Software Verification Results	11.14	②	②	②	
5	High-level requirements conform to standards.	6.3.1e	○	○	○		Software Verification Results	11.14	②	②	②	
6	High-level requirements are traceable to system requirements.	6.3.1f	○	○	○	○	Software Verification Results	11.14	②	②	②	②
7	Algorithms are accurate.	6.3.1g	●	●	○		Software Verification Results	11.14	②	②	②	

LEGEND:

● The objective should be satisfied with independence.

○ The objective should be satisfied.

Blank Satisfaction of objective is at applicant's discretion.

① Data satisfies the objectives of Control Category 1 (CC1).

② Data satisfies the objectives of Control Category 2 (CC2).

Table A-4
Verification of Outputs of Software Design Process

	Objective		Applicability by SW Level				Output		Control Category by SW level				
	Description	Ref.	A	B	C	D	Description	Ref.	A	B	C	D	
1	Low-level requirements comply with high-level requirements.	6.3.2a	●	●	○		Software Verification Results	11.14	②	②	②		
2	Low-level requirements are accurate and consistent.	6.3.2b	●	●	○		Software Verification Results	11.14	②	②	②		
3	Low-level requirements are compatible with target computer.	6.3.2c	○	○			Software Verification Results	11.14	②	②			
4	Low-level requirements are verifiable.	6.3.2d	○	○			Software Verification Results	11.14	②	②			
5	Low-level requirements conform to standards.	6.3.2e	○	○	○		Software Verification Results	11.14	②	②	②		
6	Low-level requirements are traceable to high-level requirements.	6.3.2f	○	○	○		Software Verification Results	11.14	②	②	②		
7	Algorithms are accurate.	6.3.2g	●	●	○		Software Verification Results	11.14	②	②	②		
8	Software architecture is compatible with high-level requirements.	6.3.3a	●	○	○		Software Verification Results	11.14	②	②	②		
9	Software architecture is consistent.	6.3.2b	●	○	○		Software Verification Results	11.14	②	②	②		
10	Software architecture is compatible with target computer.	6.3.3c	○	○			Software Verification Results	11.14	②	②			
11	Software architecture is verifiable.	6.3.3d	○	○			Software Verification Results	11.14	②	②			
12	Software architecture conforms to standards.	6.3.3e	○	○	○		Software Verification Results	11.14	②	②	②		
13	Software partitioning integrity is confirmed.	6.3.3f	●	○	○	○	Software Verification Results	11.14	②	②	②	②	

LEGEND:

● The objective should be satisfied with independence.

○ The objective should be satisfied.

Blank Satisfaction of objective is at applicant's discretion.

① Data satisfies the objectives of Control Category 1 (CC1).

② Data satisfies the objectives of Control Category 2 (CC2).

Table A-5
Verification of Outputs of Software Coding & Integration Processes

Objective		Ref.	Applicability by SW Level				Output		Ref.	Control Category by SW level			
Description			A	B	C	D	Description			A	B	C	D
1 Source Code complies with low-level requirements.	6.3.4a	● ● ○					Software Verification Results	11.14	② ② ②				
2 Source Code complies with software architecture.	6.3.4b	● ○ ○					Software Verification Results	11.14	② ② ②				
3 Source Code is verifiable.	6.3.4c	○ ○					Software Verification Results	11.14	② ②				
4 Source Code conforms to standards.	6.3.4d	○ ○ ○					Software Verification Results	11.14	② ② ②				
5 Source Code is traceable to low-level requirements.	6.3.4e	○ ○ ○					Software Verification Results	11.14	② ② ②				
6 Source Code is accurate and consistent.	6.3.4f	● ○ ○					Software Verification Results	11.14	② ② ②				
7 Output of software integration process is complete and correct.	6.3.5	○ ○ ○					Software Verification Results	11.14	② ② ②				

LEGEND:

- The objective should be satisfied with independence.
- The objective should be satisfied.
- Blank Satisfaction of objective is at applicant's discretion.
- ① Data satisfies the objectives of Control Category 1 (CC1).
- ② Data satisfies the objectives of Control Category 2 (CC2).

	Objective		Applicability by SW Level				Output		Control Category by SW level			
	Description	Ref.	A	B	C	D	Description	Ref.	A	B	C	D
1	Executable Object Code complies with high-level requirements.	6.4.2.1	○	○	○	○	Software Verification Cases and Procedures	11.13	①	①	②	②
		6.4.3					Software Verification Results	11.14	②	②	②	②
2	Executable Object Code is robust with high-level requirements.	6.4.2.2	○	○	○	○	Software Verification Cases and Procedures	11.13	①	①	②	②
		6.4.3					Software Verification Results	11.14	②	②	②	②
3	Executable Object Code complies with low-level requirements.	6.4.2.1	●	●	○		Software Verification Cases and Procedures	11.13	①	①	②	
		6.4.3					Software Verification Results	11.14	②	②	②	
4	Executable Object Code is robust with low-level requirements.	6.4.2.2	●	○	○		Software Verification Cases and Procedures	11.13	①	①	②	
		6.4.3					Software Verification Results	11.14	②	②	②	
5	Executable Object Code is compatible with target computer.	6.4.3a	○	○	○	○	Software Verification Cases and Procedures	11.13	①	①	②	②
							Software Verification Results	11.14	②	②	②	②

LEGEND:

- The objective should be satisfied with independence.
- The objective should be satisfied.
- Blank Satisfaction of objective is at applicant's discretion.
- ① Data satisfies the objectives of Control Category 1 (CC1).
- ② Data satisfies the objectives of Control Category 2 (CC2).

Table A-7
Verification of Verification Process Results

Objective		Applicability by SW Level				Output		Control Category by SW level			
Description	Ref.	A	B	C	D	Description	Ref.	A	B	C	D
1 Test procedures are correct.	6.3.6b	●	○	○		Software Verification Cases and Procedures	11.13	②	②	②	
2 Test results are correct and discrepancies explained.	6.3.6c	●	○	○		Software Verification Results	11.14	②	②	②	
3 Test coverage of high-level requirements is achieved.	6.4.4.1	●	○	○	○	Software Verification Results	11.14	②	②	②	②
4 Test coverage of low-level requirements is achieved.	6.4.4.1	●	○	○		Software Verification Results	11.14	②	②	②	
5 Test coverage of software structure (modified condition/decision) is achieved.	6.4.4.2	●				Software Verification Results	11.14	②			
6 Test coverage of software structure (decision coverage) is achieved.	6.4.4.2a 6.4.4.2b	●	●			Software Verification Results	11.14	②	②		
7 Test coverage of software structure (statement coverage) is achieved.	6.4.4.2a 6.4.4.2b	●	●	○		Software Verification Results	11.14	②	②	②	
8 Test coverage of software structure (data coupling and control coupling) is achieved.	6.4.4.2c	●	●	○		Software Verification Results	11.14	②	②	②	

LEGEND:

● The objective should be satisfied with independence.

○ The objective should be satisfied.

Blank Satisfaction of objective is at applicant's discretion.

① Data satisfies the objectives of Control Category 1 (CC1).

② Data satisfies the objectives of Control Category 2 (CC2).

Table A-8
Software Configuration Management Process

Objective		Applicability by SW Level				Output		Control Category by SW level				
Description	Ref.	A	B	C	D	Description	Ref.	A	B	C	D	
1 Configuration items are identified.	7.2.1	○	○	○	○	SCM Records	11.18	②	②	②	②	
2 Baselines and traceability are established.	7.2.2	○	○	○	○	Software Configuration Index	11.16	①	①	①	①	
						SCM Records	11.18	②	②	②	②	
3 Problem reporting, change control, change review, and configuration status accounting are established.	7.2.3	○	○	○	○	Problem Reports	11.17	②	②	②	②	
	7.2.4					SCM Records	11.18	②	②	②	②	
	7.2.5											
	7.2.6											
4 Archive, retrieval, and release are established.	7.2.7	○	○	○	○	SCM Records	11.18	②	②	②	②	
5 Software load control is established.	7.2.8	○	○	○	○	SCM Records	11.18	②	②	②	②	
6 Software life cycle environment control is established.	7.2.9	○	○	○	○	Software Life Cycle Environment Configuration Index	11.15	①	①	①	②	
						SCM Records	11.18	②	②	②	②	

LEGEND:

- The objective should be satisfied with independence.
- The objective should be satisfied.
- Blank Satisfaction of objective is at applicant's discretion.
- ① Data satisfies the objectives of Control Category 1 (CC1).
- ② Data satisfies the objectives of Control Category 2 (CC2).

Note: (1) Although the software configuration management objectives of section 7 do not vary with software level, the control category assigned to the software life cycle data may vary.

(2) The objectives of section 7 provide a sufficient integrity basis in the SCM process activities without the need for the independence criteria.

Table A-9
Software Quality Assurance Process

Objective		Applicability by SW Level				Output		Control Category by SW level			
Description	Ref.	A	B	C	D	Description	Ref.	A	B	C	D
1 Assurance is obtained that software development and integral processes comply with approved software plans and standards.	8.1a	●	●	●	●	Software Quality Assurance (SQA) Records	11.19	②	②	②	②
2 Assurance is obtained that transition criteria for the software life cycle processes are satisfied.	8.1b	●	●			SQA Records	11.19	②	②		
3 Software conformity review is conducted.	8.1c 8.3	●	●	●	●	SQA Records	11.19	②	②	②	②

LEGEND:

● The objective should be satisfied with independence.

○ The objective should be satisfied.

Blank Satisfaction of objective is at applicant's discretion.

① Data satisfies the objectives of Control Category 1 (CC1).

② Data satisfies the objectives of Control Category 2 (CC2).

Table A-10
Certification Liaison Process

Objective		Applicability by SW Level				Output		Control Category by SW level			
Description	Ref.	A	B	C	D	Description	Ref.	A	B	C	D
1 Communication and understanding between the applicant and the certification authority is established.	9.0	○	○	○	○	Plan for Software Aspects of Certification	11.1	①	①	①	①
2 The means of compliance is proposed and agreement with the Plan for Software Aspects of Certification is obtained.	9.1	○	○	○	○	Plan for Software Aspects of Certification	11.1	①	①	①	①
3 Compliance substantiation is provided.	9.2	○	○	○	○	Software Accomplishment Summary	11.20	①	①	①	①
						Software Configuration Index	11.16	①	①	①	①

LEGEND:

● The objective should be satisfied with independence.

○ The objective should be satisfied.

Blank Satisfaction of objective is at applicant's discretion.

① Data satisfies the objectives of Control Category 1 (CC1).

② Data satisfies the objectives of Control Category 2 (CC2).