metrowerks

CodeTEST Tool Qualification for DO-178B

Revision 1.0

This paper describes the approach taken for the qualification of the CodeTEST verification tool for use in an airborne product developed according to DO-178B

Last Modified 11/19/2001 9:35 AM

Revision History

Revision	Description	Date
1.0	First Draft.	11/15/2001

CONTENTS

1. About This Document

1.1 Introduction

This paper describes the approach taken for the qualification of the CodeTEST verification tool for use in an airborne product developed according to DO-178B.

This paper presents an overview of the FAA certification process for airborne software in order to demonstrate how the CodeTEST qualification approach fits into that process. The discussion will include the objectives for DO-178B certification and answers questions as to when and why tool qualification is needed. This paper will give details on how Applied Microsystems' approach to tool qualification satisfies these qualification objectives.

This document will focus on the qualification of software verification tools and their role and integration in the certification of airborne software.

1.2 Related Documents

- *Software Considerations in Airborne System and Equipment Certificaton-RTCA/DO-178B*, RTCA Inc., Washington D.C., December 1992
- FAA Notice N8110.91, *Guidelines for the Qualification of Software Tools Using RTCA/DO-178B*, Jan 16, 2001

  http://av-info.faa.gov/software/Policy%20&%20Guidance/N8110_91.pdf
- FAA Advisory Circular AC 25-1309-1A and/or the Joint Aviation Authorities AMJ 25-1309 FAA AC-20-115B
- Cem Kaner et al, *Testing Computer Software, Second Edition*, Massachusetts, International Thomson Computer Press, 1993
- Myers, G.J., *Software Reliability: Principles and Practices*. New York: John Wiley & Sons. 1976

1.3 Authors

Any comments about this document should be directed to:

Nat Hillary
Applied Microsystems Corporation
5020 148th AVE. N.E.
Redmond, WA 98052
Voice: (425) 882-5259
email: nath@amc.com

Tammy M. Reeve
Contract DER
Patmos Engineering Services, Inc.
tammy@patmos-eng.com

2. FAA Certification and DO-178 Overview

Whenever an avionics system is developed for commercial or military aircraft the safety of the crew and occupants is a factor in the development of the system. To this end, the FAA has established various procedures to certify that the avionics system meets their safety critical objectives.

At the very highest level, the FAA requires that the aircraft be certified and included in this certification is the avionics systems. Within the avionics system, the software must be certified and guidelines are given to how and when the software must be certified.

The *Code of Federal Regulations* (CFR) is a codification of the general and permanent rules published in the *Federal Register* by the Executive departments and agencies of the Federal Government of the USA. Title 14 CFR defines the rules that effect the approval of aircraft and aircraft equipment by the Federal Aviation Administration (FAA). The rules in Title 14 CFR are also known as the Federal Aviation Regulations (FAR).

The FAA, through Title 14, provides four different approaches to the certification of avionics as defined below:

- Design approvals under the Type Certificate (TC) or Supplemental Type Certificate (STC) approval processes
- Design approvals under the Technical Standard Order (TSO) approval process
- Installation approvals for initial (new) avionics following a TSO approval
- Installation approvals using the FAA Form 337 Process

This paper does not attempt to define these approaches but presents them as an overview as to how DO-178B relates to Avionics certification.

Whichever approach to certification is chosen, when a new system development is initiated, the developer must coordinate with the FAA Aircraft Certification Office (ACO) to get approval of one of the above Certification Plans and acceptance of the preliminary system safety assessment (discussed in Section 3).

The Certification Plan is the document that establishes the criticality of the system and it’s various components. This determines the level of design assurance required by the regulations. It also establishes the software approval levels as required by the software development standard RTCA Inc. DO-178B document. System criticality and the relationship to software levels will be discussed in the following sections.

The rules in Title 14 CFR do not reference software directly. The FAA issued Advisory Circular AC 20-115B in 1993 recognizing DO-178B as means of demonstrating compliance to the Federal Aviation Regulations for the software aspects of airborne systems and equipment.

Figure 1 illustrates the relationship between the various certifications and qualifications necessary for the FAA. Within the certification of the aircraft, the avionics software must be certified. If the developer of the avionics software is to use tools to achieve certification, the tools must be FAA qualified to ensure that the certification objectives are met. In short, the tool qualification is a subset of the avionics certification, which in turn is a subset of the aircraft certification.

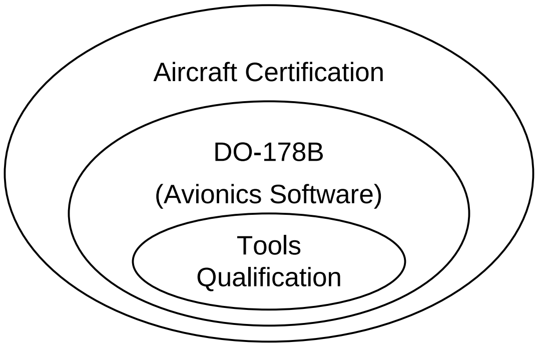

Figure 0 Relationship between certification, DO-178B and Qualification

The remainder of this paper will concentrate on the use of Coverage Analysis tools as a Software Verification tool in the development of avionic systems. The purpose of Coverage Analysis will be discussed, along with the specific coverage objectives required by DO-178B.

3. System Safety Assessments and Software Levels

As part of the certification process, the system safety assessment process is used to determine and categorize the failure conditions and criticality of an avionic system. During this process, the system design is analyzed to identify additional system requirements aimed at improving system safety. Failure conditions are identified, along with an assessment of whether the system should be immune from the failure or respond to it. The objective is to identify requirements that will preclude or limit the effect of faults, and may result in fault detection and fault tolerance requirements being added. The failure condition category of a system is established by determining the severity of failure conditions on the aircraft and its occupants. An error in software may cause a fault that contributes to an avionic system failure condition. Thus, the level of software integrity necessary for safe operation is related to the system failure conditions.

The failure condition categories used by DO-178B (derived from document AC 25-1309-1A from the FAA/and document AMJ 25-1309 from the JAA) are:

A. **Catastrophic**: Failure conditions that would prevent continued safe flight and landing.

B. **Hazardous/Severe-Major**: Failure conditions which would reduce the capability of the aircraft or the ability of the crew to cope with adverse operating conditions to the extent that there would be:

- A large reduction in safety margins or functional capabilities,
- Physical distress or higher workload such that the flight crew could not be relied on to perform their tasks accurately or completely, or
- Adverse effects on occupants including serious or potentially fatal injuries to a small number of those occupants.

C. **Major**: Failure conditions which would reduce the capability of the aircraft or the ability of the crew to cope with adverse operating conditions to the extent that there would be, for example, a significant reduction in safety margins or functional capabilities, a significant increase in crew workload or in conditions impairing crew efficiency, or discomfort to occupants, possibly including injuries.

D. **Minor**: Failure conditions which would not significantly reduce aircraft safety, and which would involve crew actions that are well within their capabilities. Minor failure conditions may include, for example, a slight reduction in safety margins or functional capabilities, a slight increase in crew workload, such as routine flight plan changes, or some inconvenience to occupants.

E. **No Effect**: Failure conditions that do not affect the operational capability of the aircraft or increase crew workload.

3.1 Coverage required for each type of software level

A DO-178B criticality level is assigned to the software under development based on the contribution that the software may make to potential failure conditions, as determined by the system safety assessment process. The software criticality level implies that the level of effort required to show compliance with certification requirements varies with the failure condition category. It follows then that testing of the airborne software has two objectives as defined in section 6.4 of DO-178B:

1. Demonstrate that the software satisfies the certification requirement applicable to the program
2. Demonstrate that the software testing verifies with a high degree of confidence that the errors, which could lead to unacceptable failure conditions, as determined by the system safety assessment process, have been removed.

Table 1 shows the relationship between this system safety definition and the software criticality level and how it relates to the verification objectives for structural coverage as defined by DO-178B section 6.4.4.2.

Table 1 Software Level vs. Structural Coverage Level

Software Criticality Level	Definition	Associated Structural Coverage Level objectives, (DO-178B 6.4.4.2 & table A-7)
Level A	Software that could cause or contribute to the failure of the system resulting in a catastrophic failure condition.	Modified Condition/ Decision Coverage, Decision Coverage & Statement Coverage
Level B	Software that could cause or contribute to the failure of the system resulting in a hazardous or severe-major failure condition.	Decision Coverage & Statement Coverage
Level C	Software that could cause or contribute to the failure of the system resulting in a major failure condition.	Statement Coverage
Level D	Software that could cause or contribute to the failure of the system resulting in a minor failure condition.	None required
Level E	Software that could cause or contribute to the failure of the system resulting in no effect on the system.	None Required

3.2 DO-178B Verification Objectives

The verification objectives for DO-178B are set in place to detect and report errors that may have been introduced during the software development processes. Software verification objectives are satisfied through a combination of reviews and analyses, the development of test cases and procedures, and the subsequent execution of those test procedures.

Reviews and analysis provide an assessment of the accuracy, completeness, and verifiability of the software requirements, software architecture, and source code. Most software code is written in a high level language such as C, C++ or Ada, and the coverage achieved by any given test is usually measured against high-level source code (also referred to as Structural Coverage). The development of test cases may provide further assessment of the internal consistency and completeness of the requirements. The execution of the test procedures provides a demonstration of compliance with the requirements. Software test cases should be based primarily on the software requirements and developed to reveal potential errors.

3.2.1 Structural Coverage Analysis Objectives

Software coverage analysis* is used to determine which requirements were not tested. This is supported by the structural coverage analysis objectives required by DO-178B that are intended to determine what software structures (e.g. statements or decisions) were not exercised as a result of these verification activities. This, in turn, reveals requirements that may have been in error, tests that were lacking adequate coverage for these structures, or dead code. Structural coverage analysis is performed to the degree required by the criticality of the software (see Table 1). Structural coverage analysis may be performed on the source code; unless the software level is A and the compiler generates object code that is not directly traceable to source code statements. Additional verification should then be performed on the object code to establish the correctness of such generated code sequences.

3.2.1.1 SC, DC, MC/DC

DO-178B defines Statement Coverage, Decision Coverage and Modified Condition/Decision Coverage as follows:

Modified Condition/Decision Coverage (MC/DC) - Every point of entry and exit in the program has been invoked at least once, every condition in a decision in the program has taken all possible outcomes at least once, every decision in the program has taken on all possible outcomes at least once, and each condition in a decision has been shown to independently affect that decision's outcome.

Decision Coverage (DC) - Every point of entry and exit in the program has been invoked at least once and every decision in the program has taken on all possible outcomes at least once.

Statement Coverage (SC) - Every statement in the program has been invoked at least once.

* One perspective on why Coverage Analysis metrics are so powerful is offered by an observation made by G. J. Myers in 1976. In considering the software-testing ideal – 100% coverage of every software execution path, Myers described a 100-line program that had 10^{18} unique paths. For comparative purposes, he noted that the universe is only about $4x10^{17}$ seconds old. With this observation, Myers demonstrated that complete software execution path testing is impossible, so an approximation alternative and another metric are required to assess testing completeness. Coverage Analysis has been accepted to be an excellent metric for assessing testing effectiveness.

4. COTS Software Verification Tools, DO-178B and Qualification

Software development, and especially testing, can be a very repetitive and human-labor intensive process, often resulting in errors and high costs. For these reasons various tools have been developed to automate portions of this process. If the tools are dependable, then productivity improvements and fewer in-service errors will be realized. In order to certify systems developed by tools, the FAA, FAA Designated Engineering Representative's (DER's), and applicants need to obtain confidence that these tools are dependable – this is done through the tool qualification process. DO-178B Section 12.2 and FAA Notice N8110.91 were designed to provide criteria for establishing which tools require additional confidence and the criteria and data needed to establish that confidence.

When a tool is used as part of the development or verification of an airborne product, the effect of that tool must be assessed to determine if qualification is required. Not all software tools need to be qualified.

4.1 When is tool qualification necessary?

According to DO-178B Section 12.2, qualification of a tool is needed only when processes described in DO-178B are eliminated, reduced, or automated by the use of that tool without its output being verified as specified in DO-178B Section 6. This means that if the results of the tool are being relied on to supply the sole evidence that one or more objectives are satisfied, the tool is required to be qualified. If the output of the tool is verified by some other means, then there is no need to qualify the tool.

This all boils down to three questions that must be asked to determine whether tool qualification is required for a Software Verification tool:

1. Can the tool allow an existing error to remain undetected?
2. Will the tool's output not be verified per section 6 of DO-178B?
3. Are processes of DO-178B eliminated, reduced, or automated?

If the answer to all of these questions is 'yes', then tool qualification is required.

4.2 Verification tool vs. Development Tool

There are two types of tools defined in DO-178B, Verification and Development Tools. DO-178B defines development tools as "tools whose output is part of airborne software and thus can introduce errors." and Verification tools as "tools that cannot introduce errors, but may fail to detect them."

4.3 Qualification Requirements

Section 12.2.2 of DO-178B states that verification tools should be qualified by "demonstration that the tool complies with its Tool Operation Requirements under normal operation conditions." Additional guidance in Notice N8110.91 states:

> "For verification tool qualification, the Tool Operational Requirements should be documented and available to the FAA (reference DO-178B, Section 12.2.3.2). The requirements for the Tool Operational Requirements data are discussed in Section 6d of this notice.

> (c) Data that shows that all of the requirements in the Tool Operational Requirements have been verified should also be documented and available for FAA review. Sufficient verification data is needed to demonstrate normal operation only and will vary depending on the complexity of the tool, the purpose of the tool, and how the tool is used. This verification data may be packaged in any document deemed acceptable by the applicant.
>
> (d) An entry summarizing the results of the verification tool qualification should be included in the Software Accomplishment Summary (SAS). The SAS should be submitted to the FAA. This allows the ACO engineer to approve the results of the verification data and is evidence of the tool's qualification status.
>
> NOTE: The applicant may choose to provide a separate Tool Qualification Plan and Tool Accomplishment Summary referenced by entries in the PSAC and the SAS for software verification tools. Entries are still required in the PSAC and SAS. This is an acceptable approach with the added benefit of providing the ability to reference a data package for reuse in subsequent certifications or in different certifications where the usage of the tool can be shown to be identical."

FAA Notice N8110.91 was written to address the misinterpretations and inconsistent application of the DO-178B tool qualification process. The issues that it helps to clarify for Software Verification tools include:

- When a tool should be qualified.
- Justification for the different qualification criteria.
- Which criteria to apply to Software Verification tools.
- Data to be produced for software verification tools.
- Acceptance criteria for tool operational requirements.
- Tool determinism.
- Tool partitioning assurance and evidence.
- Tool configuration control.

In clarifying the intent of DO-178B, Notice N8110.91 encompasses an approach that may be used not only for the initial qualification of a Software Verification tool in a specified environment and system, but also for the reuse of the resulting qualification data in subsequent qualifications of the tool. Subsequent qualifications may be for different environments and systems, as well as new versions of the Software Verification tool. This approach was presented to the FAA Aircraft Certification Office (ACO) and coordinated with FAA National Resource Specialist Leanna Rierson. In April of 2000 this approach was approved.

What is meaningful in this additional guidance is the final note that states that qualification data packages may be reused where the usage of the tool can be shown to be identical. This means that evidence of past qualifications may be used to achieve future qualifications.

4.4 Qualification process

The qualification of a Software Verification tool follows a fixed process. The process described here is in keeping with the requirements of Notice N8110.91, where a separate Tool Qualification Plan and Tool Accomplishment Summary are referenced by the PSAC and SAS for Software Verification tools. In summary, the qualification process for a Software Verification tool takes the following steps:

1. The avionics developer creates and submits to the Certification Authority the *Plan for Software Aspects of Certification* (PSAC) document. This document includes a reference to the *Tool Qualification Plan* and *Tool Accomplishment Summary* documents.
2. The *Tool Operational Requirements* document for verification tool is submitted to the Certification Authority. This document will include references to qualification tests conducted to prove that the Software Verification tool operates correctly and reliably in the development environment.
3. The avionics developer creates *Software Accomplishment Summary* document that references the Software Verification tool coverage reports generated during testing. This document will also include a reference to the qualification tests conducted to prove that the Software Verification tool operates correctly and reliably in the development environment.

The tool qualification process is complete when the certification authority approves the *Tool Qualification Plan* and *Tool Accomplishment Summary* documents as evidence that the software verification tool complies with its Tool Operation Requirements under normal operation conditions.

5. Qualifying CodeTEST as a Software Verification tool

Applied Microsystems CodeTEST is a Structural Coverage Analysis tool that can be qualified for use in an avionics process. In keeping with the approach of Notice N8110.91, each new CodeTEST qualification can utilize the data from previously successful qualifications. This section describes the qualification data package that has been prepared by AMC to streamline this process, the variables that affect the use of this data, and the process that is used to achieve CodeTEST qualification.

CodeTEST operates by instrumenting source code and then measuring the structural coverage achieved while executing the software in its target environment. From these measurements, CodeTEST can generate the necessary structural coverage analysis reports for Statement Coverage (SC), Decision Coverage (DC) and Modified Condition/Decision Coverage (MC/DC). As such, CodeTEST is a software verification tool, as defined by DO-178B. CodeTEST also satisfies the DO-178B objective of determinism; that is, that CodeTEST produces the same output for the same input data when operating in the same environment.

Because the answer to the 3 questions in section 4.1 is YES for CodeTEST, CodeTEST can be qualified for use as a Software Verification tool. To this end, AMC has prepared a qualification package to ease the qualification process.

5.1 CodeTEST qualification package

The CodeTEST qualification approach and associated qualification documentation (referred to as the qualification kit “qual kit”) encompass an approach for the initial qualification of the CodeTEST tool for use in a specified environment and system. This approach also allows for the reuse of qualification data in subsequent qualifications of the tool for different environments and systems, as well as new versions of the product, as identified in FAA Notice N8110.91.

The AMC Tool qual kit includes a *Tool Qualification Plan* and *Tool Accomplishment Summary* documents, plus a *Tool Version Description* document (*Configuration Index* document). In addition to these data, AMC has developed and maintains a Requirements Document, Verification Procedures and Results document, Traceability Matrix and Version Description Document data for CodeTEST. This facilitates the Change Impact and Regression Analysis necessary based on changes to the approved qualification variables for a given version of CodeTEST and satisfies the objectives set forth in N8110.91 for a verification tool.

The AMC approach defines the responsibilities of the FAA ACO, the Avionics Company and the Tool Vendor (AMC) in the certification process when CodeTEST qualification data is to be reused.

5.1.1 Qualification Variables

Several possible variables have been identified as being an important part of the change impact analysis needed to define how the CodeTEST Qualification Documentation Kit will be evaluated for use in subsequent certifications. The variables that are assessed for each change impact and regression analysis are:

- CodeTEST Version
- Programming language

- Compiler
- CodeTEST Host Operating System
- Target Hardware Processor
- Certification Level

5.1.2 AMC Qualfication Documentation

The AMC Tool qual kit includes a *Tool Qualification Plan* document, a *Tool Accomplishment Summary* document, and a *Tool Version Description* document (same content as DO-178B Configuration Index document). In addition, AMC's Contract DER can provide an independent audit report and form 8110-3 approving the qualification data for each unique configuration of qualification variables for an airborne customer (see Section 6 – Additional Considerations).

CodeTEST approval is integrated with the applicant's project approval depending on how CodeTEST will be used in the applicant's approval process. This can include an initial tool approval, approval of a subsequent change to the tool, or for use of the approved tool within the applicant's project. Typically, the avionics applicant responsibility is as follows:

- The applicant (avionics supplier) must specify as part of the airborne product Plan for Aspects of Certification (PSAC), the intent to use CodeTEST as a verification tool. Reference can be made to the AMC data Tool Qualification Plan and baseline qualification approach.
- The applicant (avionics supplier) must include a summary of the results of the tool qualification data in the airborne product Software Accomplishment Summary. This includes the summary of activities for evaluating the qualification variables on the avionics hardware and software platform.

Table 2 below describes the activities and responsibilities as defined for the CodeTEST reusable qualification.

Table 2 CodeTEST Qualification Process Activities Reference

Process Activity	Responsibility
Verify Qualification Variables.	AMC
Ensure CodeTEST Qualification Letter and/or Documentation and services have been ordered from AMC.	AMC & Customer
Update Tool Qualification Plan document and execute Change Impact Analysis Services.	AMC
Document intent and use of CodeTEST in PSAC.	Customer
Submit PSAC and CodeTEST Tool Qualification Plan and/or pre-approval letter to certification Authority.	Customer
Execute TaskWalk program to verify proper Hardware Target Processor connection to CodeTEST Hardware Probe.	Customer
Verify the proper configuration of the CodeTEST system in the certification system environment. Run a test with known results with and without instrumentation and verify the results are the same.	Customer
Execute the full set of requirements-based tests for the certification system on the instrumented source code (Note: the original system source code should be checked under configuration control, a copy	Customer

Process Activity	Responsibility
should be checked out for the purposes of coverage testing).	
Analyze the source code structural coverage results of the requirements-based tests documented in the CodeTEST Coverage Report to determine if the coverage goals have been achieved.	Customer
Archive the CodeTEST datafiles, coverage reports, idbs and instrumented source code once the coverage goals have been achieved by the requirements-based tests.	Customer
Execute the full set of requirements-based tests for the certification system on the un-instrumented source code and verify the results.	Customer
Document the identification of the test environment including all CodeTEST ACT configuration information, host configuration files used, instrumenter configuration.	Customer
Update Tool Qualification Accomplishment Summary document to include results of the activities defined in the Change Impact Analysis.	AMC
Create Software Accomplishment Summary and document either the use of CodeTEST qualification data and/or the use of a pre-approval letter along with information that documents the use of CodeTEST are within the bounds of the approach documented in the data provided.	Customer

6. ADDITIONAL CONSIDERATIONS

FAA Notice 8110.91 makes provisions for the use of an FAA Designated Engineering Representative (DER) as part of the tool qualification approval process. "*If the ACO engineer has delegated compliance findings for tool qualification data, DERs may approve the tool qualification data which complies with the guidance of DO-178B, Section 12.2. However, approval of alternative methods and the resultant data should be retained by the ACO engineer*."

AMC utilizes the services of a DER in the preparation, review and approval of the tool qualification data for CodeTEST. This has benefited many avionics suppliers, the FAA, and AMC, in that it provides a consistent implementation of the approved approach to the qualification of CodeTEST for multiple airborne certification programs. This gives each avionics supplier the ability to leverage the implementation of CodeTEST and its years of field service, while at the same time having project-specific approval for the use of CodeTEST on their application.

7. SUMMARY

The certification and tool qualification processes are integral in the development of an airborne system. Tool qualification is a part of the certification process and must be a part of the overall planning for an airborne system. It is important that the tools used in the development and verification of the airborne software are decided on as early as possible in the life cycle of the project. Identifying the right tools and having them in place early in the life cycle can have a significant impact on the success and on-time delivery of the system.

The CodeTEST Advanced Coverage tool provides SC, DC and MC/DC levels of Coverage Analysis, and can be qualified for use as a Software Verification tool in the development of avionic systems. AMC has prepared a qualification package of data that has been prepared in keeping with the guidelines outlined in FAA Notice N8110.91, whereby data from previous CodeTEST qualifications may be used as evidence in new qualifications. This results in a streamlined qualification process, and is a valuable resource for avionics suppliers.