Decision Point: Will Using a COTS Component Help or Hinder Your DO-178B Certification Effort?©

Timothy J. Budden
AVISTA, Incorporated

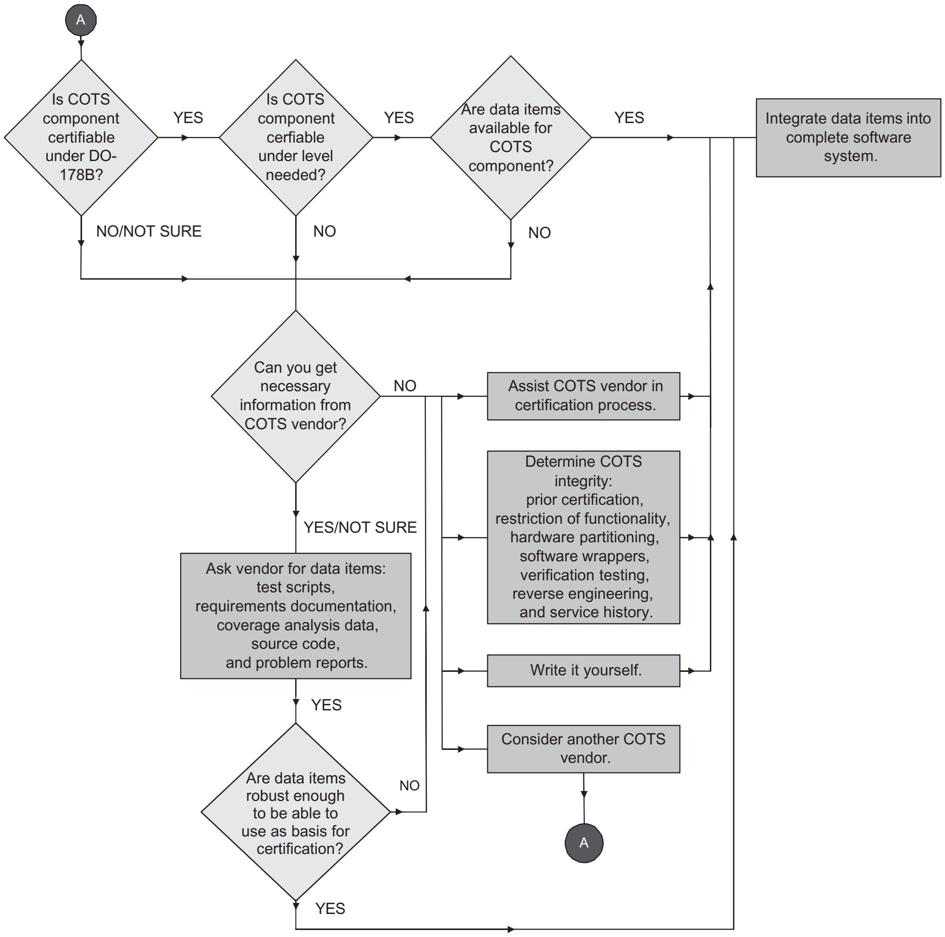
Avionics software developers today are continually challenged to cut costs and reduce time to market, without compromising the safety of their application. Many project leaders look to commercial off-the-shelf (COTS) software components as a possible means to reduce software development costs and development time. The requirements to "prove" software quality under Defense Order (DO)-178B may be difficult, but the opportunity demands consideration of COTS module integration where possible. Understand what is certifiable, how to get the right information from your vendor, and the importance of DO-178B traceability.

Nearly all embedded applications intended for avionics deployment must pass the rigorous certification guidelines developed by the Radio Technical Commission for Aeronautics, Inc. (RTCA) for use by the U.S. Federal Aviation Administration (FAA) in certifying software used in commercial aircraft. These guidelines, known as RTCA Defense Order (DO)-178B, prescribe the development and verification process for software intended for airborne systems and other equipment that must meet certain FAA criteria for airworthiness.

Generally, certification is required for airborne systems and related equipment whose failure will put human life at risk. The two regulatory bodies that primarily administer these safety-critical issues include the U.S. FAA and the Joint Aviation Authority in Europe. These agencies recognize DO-178B as an acceptable means of compliance for software approval in airborne systems.

Certification of avionics equipment is typically achieved through FAA authorization of a type certificate, parts manufacturer approval, or a technical standard order. Systems are categorized by DO-178B as Level A through Level E, based on their criticality in supporting safe aircraft flight. Level A is the most critical, as a failure of such a system could result in a catastrophic failure condition for the aircraft. Level E is the least critical, as a failure of such a system has no effect on the operational capability of the aircraft or pilot workload.

Figure 1: *Decision Tree to Determine if COTS Component Is Certifiable*


A
Is COTS component certifiable under DO-178B?
YES
Is COTS component cerfiable under level needed?
YES
Are data items available for COTS component?
YES
Integrate data items into complete software system.
NO/NOT SURE
NO
NO
Can you get necessary information from COTS vendor?
NO
Assist COTS vendor in certification process.
Determine COTS integrity: prior certification, restriction of functionality, hardware partitioning, software wrappers, verification testing, reverse engineering, and service history.
YES/NOT SURE
Ask vendor for data items: test scripts, requirements documentation, coverage analysis data, source code, and problem reports.
Write it yourself.
YES
Consider another COTS vendor.
Are data items robust enough to be able to use as basis for certification?
NO
A
YES

Is the COTS Component DO-178B Certifiable?

The conundrum regarding whether or not commercial off-the-shelf (COTS) modules will help or hinder the developer is better understood in the context of system certification. DO-178B certification requires applying stringent processes for all software, including COTS components that ultimately make up the end software system. This includes generating software life-cycle data items that support the entire software system, including any COTS software that may be incorporated into the application.

It is important to note that although a software component may have been previously included in other systems that were certified under DO-178B, it does not necessarily follow that the software component will be certifiable in the new system. This complicates the COTS decision. How does the software developer determine whether to incorporate a COTS component that claims to be certifiable or is believed to be certifiable?

How a software component is used is more important than its prior certification history. It is not possible for COTS vendors to receive standalone certification for particular software components they supply and to have that component *automatically* be certified when incorporated into an application. Moreover, COTS vendors who claim to be DO-178B certifiable may not be certifiable to the level (A through E) that is required. Regardless, while it is not possible to certify a COTS module in isolation, it is possible to package that COTS component in a form that facilitates certification by a systems developer.

Software Life-Cycle Data Items	Description	DO-178B Level				
		A	B	C	D	E
Planning						
Plan for Software Aspects of Certification	Used by certification authority to determine whether applicant is proposing a software life cycle commensurate with the rigor required for the level of software being developed.	XX[1]	XX	XX	X	
Software Project Development Plan	Includes objectives, standards, and software life cycles to be used in the software development process.	XX	XX	XX	X	
Software Verification Plan	Describes the verification procedures to satisfy the software verification process objectives.	XX	XX	XX	X	
Software Configuration Management Plan	Establishes methods to be used to achieve the objectives of the software configuration process throughout the software life cycle.	XX	XX	XX	X	
Software Quality Assurance Plan	Describes the methods used to achieve the objectives of the software quality assurance process.	XX	XX	XX	X	
Standards						
Software Requirements Standards	Defines the methods, rules, and tools to be used to develop the high-level requirements.	X	X	X		
Software Design Standards	Defines the methods, rules, and tools to be used to develop the software architecture and low-level requirements.	X	X	X		
Software Code Standards	Defines the methods, rules, and tools to be used to code the software.	X	X	X		
Project Development						
Software Requirements Data	Describes the high-level requirements, including derived requirements.	X	X	X	X	
Design Description	Describes the software architecture and low-level requirements that will satisfy the software high-level requirements.	X	X	X	X	
Source Code	Consists of code written in source language(s) and the compiler instructions for generating the object code from the Source Code, and link and loading data.	X	X	X	X	
Executable Object Code	Consists of a form of Source Code that is directly usable by the central processing unit of the target computer and is the software that is loaded into the hardware or system.	X	X	X	X	
Software Verification						
Software Verification Cases and Procedures	Details how the software verification process activities are implemented.	XX[2]	X	X	X	
Software Verification Results	Results that are produced by the software verification process activities.	XX	X	X	X	
Additional Data Items Spanning Entire Life Cycle						
Software Life-Cycle Environment Configuration Index	Identifies the configuration of the software life-cycle environment. The index aids reproduction of the hardware and software life-cycle environment.	X	X	X	X	
Software Configuration Index	Identifies the configuration of the software product.	X	X	X	X	
Problem Reports	Reports identify and record resolution to software product anomalous behavior, process non-compliance with software plans and standards, and deficiencies in software life-cycle data.	X	X	X	X	
Software Configuration Management						
Software Configuration Management Records	Results of the software configuration management process activities.	X	X	X	X	
Software Quality Assurance						
Software Quality Assurance Records	Results of the software quality assurance process activities.	XX[3]	XX	X	X	
Software Aspects for Certification						
Software Accomplishment Summary	Used as the primary data item for showing compliance with the Plan for Software Aspects for Certification.	X	X	X	X	

[1] The number of X's indicates the level of rigor and detail expected for that specific data item for that level of certification.
[2] Increasing in rigor and independence
[3] Independence for all four levels

Table 1: *Data Items Necessary for DO-178B Certification*

Figure 1 displays a decision tree that suggests the types of questions to ask a COTS vendor when deciding whether or not to use a COTS component as part of your avionics application. The remainder of this article focuses on the details of each stage of inquiry as reflected in Figure 1.

The ideal situation is to purchase a COTS component that provides all the necessary life-cycle data items to support DO-178B certification. However, it is by no means a common option among COTS vendors. You may need to do a bit more research to determine if you and the COTS vendor can work together to satisfy DO-178B requirements to get the necessary life-cycle data items for certification for your entire avionics application.

Get the Right Information From the COTS Vendor

Knowing what data items to get from your potential COTS vendor will depend upon your overall system approach to certification. Moreover, the level of detail necessary for certain data items varies based on the level of DO-178B software certification to which your avionics software application must comply. The process of obtaining the necessary information to support certification from the COTS vendor requires a formal business relationship between your companies. At a minimum, you should expect that the COTS vendor would work closely with your system developers to ensure acceptance of the COTS component within your avionics system.

Table 1 outlines the data items from your software life-cycle process that are expected as part of the overall certification process. These data items are as follows:

- Planning documents that include Plan for Software Aspects for Certification, Software Project Development Plan, Software Verification Plan, Software Configuration Management Plan, and Software Quality Assurance Plan.
- Standards documents that include Software Requirements Standards, Software Design Standards, and Software Code Standards.
- Project development data that include software requirements data, design description, source code, and executable object code.
- Software verification that includes Software Verification Cases and

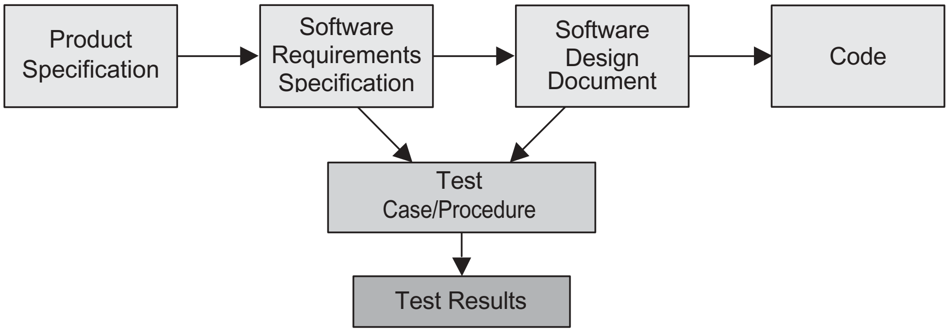


Figure 2: *Traceability Flow*

Procedures, and Software Verification Results.

- Additional life-cycle data items that span the entire life cycle, including Software Life-Cycle Environment Configuration Index, Software Configuration Index, and Problem Reports.
- Configuration management records.
- Software quality assurance records.
- Software accomplishment summary.

Detailed descriptions of these data items can be found in various official publications governing DO-178B development such as RTCA DO-178B, "Software Considerations in Airborne Systems and Equipment Certification," Dec. 1, 1992. As Table 1 suggests, different certification levels may require different degrees of detail or completeness in each data item. Understanding the certification level that you plan for your application is a necessary precursor to the dialogue with your COTS vendor regarding needed data items.

Mixing and matching COTS vendor data items with your own data items can be done in a variety of ways. For example, you may wish to incorporate details of the COTS vendor's verification process into your overall Software Verification Plan. You may then decide either to include the COTS vendor's test case/procedure data into your own Software Verification Cases and Procedures document, or have it stand alone as a cases and procedures document solely for the COTS component. The key is that the processes are documented and followed, and that the data items are captured, regardless of how they are packaged.

The Importance of Traceability and Independence

Traceability is a well-defined manner to objectively assess the rigor applied to the development and verification of the entire system. That is, satisfying the traceability requirements of DO-178B certification involves documenting how downstream life-cycle elements link to upstream life-cycle elements. For example, design elements and test case/procedure elements must be linked to originating requirement elements.

To verify your software, DO-178B requires both a Requirements Coverage Analysis (RCA) and a Structural Coverage Analysis (SCA). The RCA requires traceability and ensures that a test case/procedure exists for every software requirement. The SCA uncovers code elements that were not covered through execution of requirements-based tests. The rigor of the SCA varies with the criticality level of the software. Having top-to-bottom traceability also facilitates regression analysis activities when change inevitably occurs. The traceability flow is shown in Figure 2.

> *"More times than not, your leadership will be demanded in helping to bridge understanding with your proposed COTS vendor of how and what is required to support the certification effort."*

Independence is also an important DO-178B topic. Independence is the separation of responsibilities that ensures the accomplishment of objective evaluation. For software verification process activities, independence is achieved when a person(s) other than the developer of the item being verified performs the verification activity; a tool(s) may be used to achieve equivalence to the human verification activity. For the software quality assurance process, independence also includes the authority to ensure corrective action.

A COTS vendor who provides data items such as requirements-based test cases/procedures may also need to prove that these tests were produced by someone other than the code developer for a given set of requirements. Independence requirements vary with the software level, but they are primarily related to verification and software quality assurance activities.

Both parties understanding these elementary concepts of traceability and independence often facilitate effective communication with your COTS vendor regarding certification. The most prevalent obstacles to incorporation of COTS modules are a lack of life-cycle data items (e.g., requirements data, design data, and test cases/procedures), traceability data, and independence. The rigor of DO-178B development is seldom adopted in commercial applications and often not understood or appreciated by embedded software developers.

What if You Can't Get the Information You Need?

In the end, the COTS vendor may be either unable or unwilling to provide the necessary data items associated with the COTS component. In this situation, you can pursue one of the following alternatives:

1. **Assist the COTS vendor in the certification process.** The COTS vendor may be interested in collaborating with you to certify your application of their product to DO-178B guidelines. This option can be a win-win situation for both parties. An example of a potentially successful business arrangement could include your company receiving the source code of the COTS module for little or no license fee in exchange for your company's assistance in working together to certify the module under DO-178B. The COTS vendor would presumably benefit from the experience gained by having a library of required life-cycle data items, as well as the promotional value of having their module(s) branded as *certifiable*.
2. **Without assistance from the vendor, determine the COTS component integrity.** One or more of the following methods may be helpful as a process to obtain the necessary information to support compliance of the COTS module with DO-178B certification:
   - Reference prior certification records in which the COTS component was approved as part of an earlier certified or qualified system application.
   - Restrict the functionality by only using a subset of functionality and certify only those functions. This

may limit the amount of information required for the certification.

  - Partition the system. This method prevents failures from noncritical functions affecting critical functions (such as implementing functions on different processors or different memory partitions).
  - Use software protection wrappers to limit the functionality exposed to the certification-targeted application. *Wrapper software* accompanies other software to improve compatibility or security such as the deactivation of unneeded functionality, and/or check the validity of parameters.
  - Analyze the COTS vendor's data items for certifiability and use/enhance as necessary.
  - Reverse engineer the COTS data items. This requires reconstructing the data, which can be a difficult task, perhaps requiring as much effort as recreating the development in-house. However, the process will produce software life-cycle data that can be reviewed and analyzed to satisfy the DO-178B objectives.
  - Reference the service history of the COTS component. This can provide previous in-service experience of the component. However, the data integrity of the service history records must be validated, which thus requires information about the problem tracking process and the software configuration management process used originally by the COTS vendor.

3. **Write the functionality in-house.** This may be your best option, especially if there are no COTS vendors that have the necessary DO-178B certifiable component(s), or who are unwilling and/or unable to provide you the necessary data items. Despite the usefulness and appeal of COTS solutions, the cost and time to develop the software or systems component in-house may be considerably less than attempting to bludgeon your way into certification of software never developed with intentions of satisfying the stringent quality concerns of DO-178B.
4. **Consider another COTS vendor.** If there are other COTS vendors that have the necessary DO-178B certifiable component(s), or are more willing and/or able to give you the necessary data items, then consider these vendors as viable alternatives. The value of working with vendors who have already committed (or are willing to commit) their energies to ensuring a DO-178B quality product should be readily apparent.

Conclusion

Certification under DO-178B is one of the most grueling development and verification processes developed, but for good reasons. There can be no compromises to software and systems quality when lives are at stake both in the air and on the ground. The requirements to *prove* software quality under DO-178B may require you to think again about your plan to incorporate a COTS module in your application. However, the opportunity to speed your time to market and improve your development productivity demands that you at least consider COTS module integration where possible.

As described in this article, the demands of DO-178B certification can be achieved with COTS modules if your vendor is a willing partner who understands the value, importance, and professionalism that is expected of DO-178B development. More times than not, your leadership will be demanded in helping to bridge understanding with your proposed COTS vendor of how and what is required to support the certification effort. The business payoff is significant for all parties, and the quality solution that results is of pride to all.

More information on RTCA DO-178B is available online at <www. rtca.org>.◆

About the Author

Timothy J. Budden is senior programs manager at AVISTA, Incorporated. His experience spans a wide range of fixed- and rotary-wing aircraft systems. He has developed a working knowledge of all software life-cycle phases and RTCA DO-178B guidelines through years of successful certification experience. Prior to joining AVISTA 12 years ago, Budden was employed by McDonnell Douglas. He has a Bachelor of Science in electrical and computer engineering from the University of Notre Dame.

AVISTA, Incorporated
P.O. Box 636
1575 U.S. Highway 151 East
Platteville, WI 53818
Phone: (608) 348-8815
Fax: (608) 348-8819
E-mail: tim.budden@avistainc.com