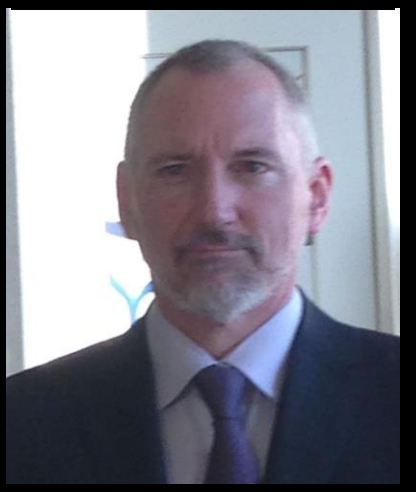
DO-330: TOOL QUALIFICATION OVERVIEW

FOR AVIONICS ENGINEERS AND MANAGERS



AUTHOR
VANCE HILDERMAN

- **BSEE, MSEE, MBA**
- ***Founder of two of the world's largest avionics development services companies***
- ***Developer of the world's first training in DO-178 and trainer of over 8,000 engineers in 45 countries in DO-178, DO-254, DO-278, and DO-200A***
- ***Primary author of the world's first, and best selling, book on DO-178 and DO-254 (available at most major bookstores worldwide)***

TECHNICAL REVIEWERS

Bunyamin Coskun,
Project Manager, Milsoft Inc

Dave Deloria,
Director of Engineering, Crane Aero.

Ron Houde,
Mannarino Systems & Software

Roger Plieske
Rohde & Schwarz

Edited by Anna Hilderman

Engineering "Tools" - Overview

Software and hardware engineering tools are computer programs that help engineers create, analyze, verify, track, modify, produce or specify the application programs being developed. Such tools and programs have been in use since nearly the beginning of computing. Tools aid the improvement of efficiency and effectiveness in the development process by automating mundane or complex operations; they also bring the level of abstraction and understanding closer to the developer. However, can tools always be trusted or must they be formally "qualified" in some cases? The answer to that important question follows, along with details of performing such tool qualification when necessary. However, it's not a black and white question, but rather a pragmatic exercise to determine "to what extent should we trust a tool, AND under what conditions do we need to formally qualify our tools?"

Today's high-reliability products utilize development tools within a variety of safety-critical applications; it is virtually impossible to accomplish the engineering development process without such tools. These tools may eliminate, reduce, or automate processes which ensure the correctness of the safety-critical application. Systems for aviation, medical, railroad, space, automotive, and military industries are developed with the assistance of development tools which may contribute to faulty operation resulting in malicious behavior of the application. The development environment can affect the design and behavior of the product and must be taken into consideration. Also, a tool used to assist in verification may be incorrect, leading to an undetected error within the product.

vance.hilderman@afuzion.com

In the aviation industry, engineering tools' potentially negative effects on avionics products must be mitigated and are regulated through the application of RTCA/DO-178C/ED-12C for software and RTCA/DO-254/ED-80 for hardware development programs respectively. And ground-based systems for CNS/ATM (Communication Navigation Systems / Air Traffic Management Systems) also have similar tool qualification needs. What do these aviation related systems have in common? They all rely upon the guidance of DO-330 for tool qualification. And in many cases per DO-330, it's not necessary to actually qualify a software tool unless the output of that tool is not otherwise verified.

What is a Tool?

Webster's dictionary defines a tool as an instrument; anything used as a means to an end, something used in the performance of an operation, anything regarded as necessary to carrying out one's occupation or profession, or a person that is used or manipulated by another (subject of another discussion, but an interesting analogy). RTCA/DO-178C, RTCA/DO-254 and FAA Order 8110.49 define a tool as a computer program used to develop, test, analyze, produce, or modify another program or its documentation. So for avionics, a tool consists of software itself used somewhere within the lifecycle of avionics systems. Consider the following common types of tools used to develop software in the figure below:

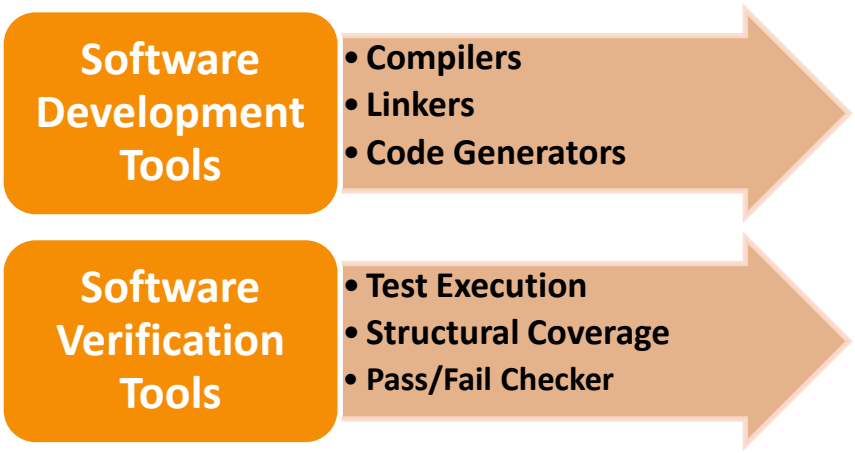


Figure 1: Common Software Engineering Tool Types

Each of the tools in the above figure plays a key role in avionics engineering and each is available ready-to-use, as "commercial off-the-shelf" (COTS) software which can be purchased directly from any of dozens of software tool vendors. Under DO-178B, tools were simply classified as "development" tools or "verification" tools. However, DO-178C does away with such a simple classification because technical advances have allowed for hybrid tools which perform verification while also reducing subsequent development activities; this is explained later herein via tool "criteria".

Tools used during engineering exist in all project phases: requirements specification, software design and code, integration, configuration management, and verification. Although it is possible to develop a safety-

vance.hilderman@afuzion.com
www.afuzion.com

critical application in the aviation industry with the use of only implementation tools (compiler, assembler, and linker), this is increasingly unlikely given the complexity and enormity of electronic systems and modern avionics in this era.

When is It Necessary to Qualify a Tool?

Tool qualification is required whenever the design assurance process(es) described in RTCA/DO-178C or RTCA/DO-254 are eliminated, reduced, or automated by the use of the tool unless the output of the tool is verified. Verification of the tool's output must be accomplished through the verification process as defined by RTCA/DO-178C Section 6. Remember, in avionics development, "Verification" has a specific meaning (as the following is not official FAA/EASA policy, the author calls it the "Hilderman Verification Equation"):

Figure 2: Hilderman Verification Equation

DO-178B Versus DO-178C Tool Qualification

Under the former DO-178B (which of course is eclipsed by DO-178C), tool qualification was addressed simply within DO-178B itself and clarified via the FAA's ubiquitous 8110.49: tools were categorized as simply one of the following:

1. **Development Tools:** *capable of inserting an error within operational flight software; or*
2. **Verification Tools***: incapable of inserting an error, but potentially capable of failing to detect an error in the flight software.*

However, DO-178B was released in 1992, in the earlier days of advanced software development, which was before associated guidelines for complex electronic hardware (DO-254) and CNS/ATM (DO-278A) were released. The all-too-brief (less than four pages) tool qualification guidelines within DO-178B were just that: often too brief. So new avionics software tool qualification guidance was needed for multiple reasons as summarized in the following figure:

vance.hilderman@afuzion.com
www.afuzion.com

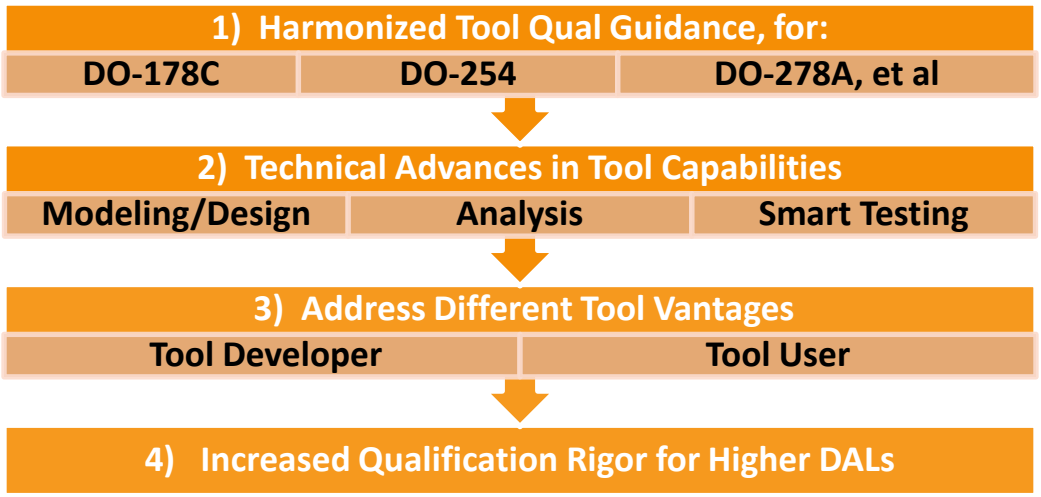


Figure 3: Reasons for DO-330 As New, Standalone Tool Qualification Guideline

As depicted above, there were four key reasons for the introduction of DO-330 "Software Tool Qualification Considerations" to provide the necessary supplementary tool qualification information in one document. Determining whether any tool needs to be qualified is accomplished by assessing the outcome of three questions regardless of the tool category.

1. Can the tool insert an error into the airborne software/hardware or fail to detect an existing error in the software/hardware within the scope of its intended usage?
2. Will the tool's output not be verified or confirmed by other verification activities as specified within for example, Section 6 of DO-178C for software?
3. Will the output of the tool be used to either meet an objective or replace an objective of RTCA/DO-178C, DO-254, DO-278A, etc.?

If the answer to all three questions is YES then the tool will most likely be required to be qualified. The figure below poses these questions via a classic flow chart. It should be noted that the answer to the first and third question is almost always "Yes", therefore the real question of tool qualification necessity normally comes down to just one question: "Will the tool output be verified?" If the answer is "No" then qualification is almost always required. A simple flow-chart for these questions is depicted below in Figure 4. To be honest, most tools can either insert an error or fail to detect an error; also most tools eliminate, reduce, or automate avionics development/verification processes. Therefore the real question to be answered in determining if a tool needs to be qualified is "Is the output of the tool otherwise verified?" If the output of an DO-178C, DO-254, or DO-278A tool is not verified then almost always that tool must be qualified. Once you determine that a tool needs to be qualified, then you determine the tool's TQL for your application and apply DO-330 objectives to that tool instance.

vance.hilderman@afuzion.com
www.afuzion.com

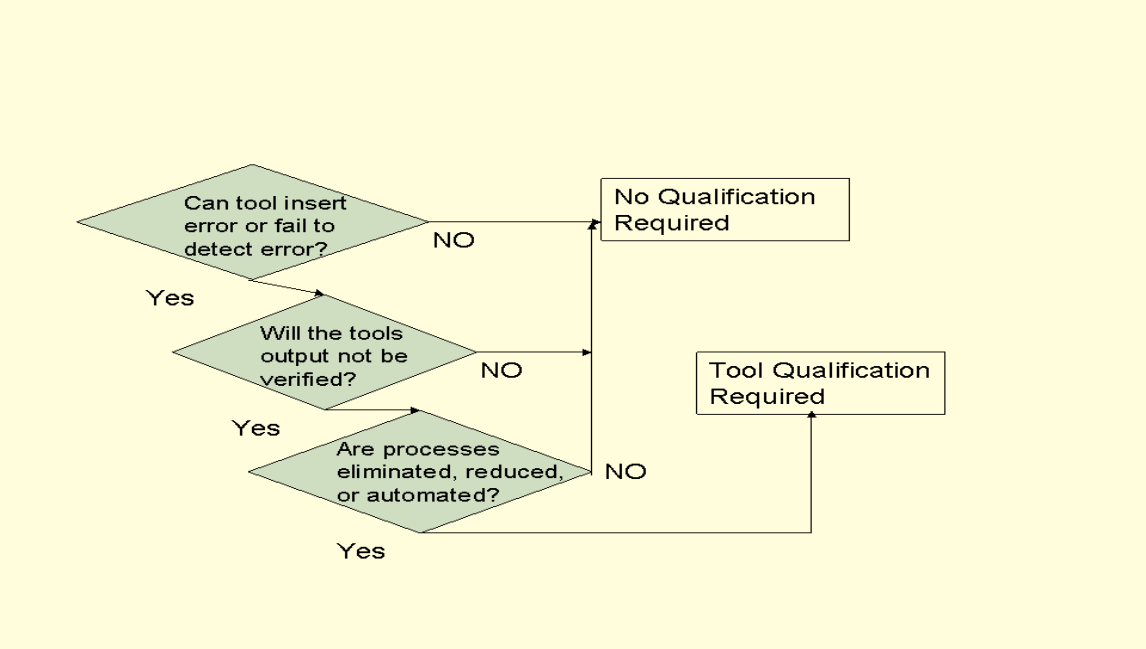


Figure 4: Simplified Flowchart to Determine Necessity of Qualifying a Tool

Why Qualification?

Flight hardware, software, and systems are normally "certified". However, tools are used in development and/or verification and the tool itself doesn't normally fly or execute onboard the aircraft during flight. However, reliance is being placed on the tool to provide evidence and output which meet certification objectives, therefore confidence must be established to prove the tool provides at least the equivalent assurance of the certification process(es) which is/are eliminated, reduced, or automated. The dependability of the tool being used must be established. Establishing the dependability of the tool and building the confidence that the tool provides at least the equivalent design assurance process for the level required is accomplished by the tool qualification process. The first step of which is to establish whether a tool needs to be qualified as described above.

Tool qualification determination and rationale, whether or not a tool requires qualification, should be established as early as possible with the certification authority and addressed in the certification planning documents. Tool assessment and evaluation should be performed during the planning phase of the project prior to proceeding with development and verification. If a tool is found not to require qualification, such agreement should be established early in order to avoid issues later in the project. Development may proceed prior to tool qualification being accomplished; however it's important to consider qualifiability to ensure qualification can be performed when required. This author has encountered numerous projects over the decades which "assumed" tool qualification for a given project was not required; subsequent results were disastrous when use of that tool's output was later disallowed when the tool was unable to be qualified.

DO-330 provides for tool qualification activities to be directly related to the potential tool impact; that impact is based upon both the category of the tool and also the DAL (Design Assurance Level) it's applied to. Therefore, DO-330 introduces five Tool Qualification Levels (TQLs) based upon three Tool Criteria.

vance.hilderman@afuzion.com
www.afuzion.com

Software DAL (Criticality Level)	Tool Criteria		
	1	2	3
Level A	TQL-1	TQL-4	TQL-5
Level B	TQL-2	TQL-4	TQL-5
Level C	TQL-3	TQL-5	TQL-5
Level D	TQL-4	TQL-5	TQL-5

Figure 5: Tool Qualification Levels: Based Upon DAL and Tool Criteria

There are three tool criteria, meaning a tool's usage is assessed to fall within one of the following three criteria categories:

1. **Criteria 1:** *A tool whose output is part of the airborne software and thus could insert an error.*
   - Example: code generation tool which automatically generates source code from models
2. **Criteria 2:** *A tool that automates verification process(es) and thus could fail to detect an error, and whose output is used to justify the elimination or reduction of:*
   a) *Verification process(es) other than that automated by the tool, or*
   b) *Development process(es) that could have an impact on the airborne software.*
   - Example: model-checking tool which verifies completeness while also checking coverage
3. **Criteria 3:** *A tool that, within the scope of its intended use, could fail to detect an error.*
   - Example: structural coverage tool that assesses code coverage

How is Tool Qualification Level (TQL) determined? Consider the preceding figure for Tool Qualification Level. If a tool's output, such as a code-generator, comprises DAL B software, then it's a Criteria 1 tool with a TQL of 2. On the other hand, if the tool is a structural coverage verification-only tool, used on any DAL, it is a Criteria 3 tool with a TQL of 5.

Which Tools Require Qualification?

vance.hilderman@afuzion.com
www.afuzion.com

Not all tools require qualification! By using the three determining questions previously discussed, it is relatively easy to establish which tools will or will not require qualification. Tools which typically reside in the requirements management, configuration management/data management, and quality management categories can generally be excluded from the tool qualification process. Why? Such tools generally do not supply output which either meet an objective or replace an objective of the Annex, or the tool's output is verified by another verification activity downstream of the tool's output. However, the assessment of the tool should be accomplished and will establish the need to qualify or not. The tools which generally require qualification fall cleanly into the Criteria 1, 2, or 3 categories. It is important to determine early if a tool needs to be qualified, and if so, its associated TQL. Recommendation: even if the tool assessment is thus shown to preclude qualification, cite the tool within the certification planning documents along with the rationale for why tool qualification is not required: be honest, be up-front to prevent problems downstream.

Compilers, assemblers, and linkers are typically Criteria 1 tools, but their output is often examined by another verification activity (i.e. review or testing). Therefore, they typically do not require tool qualification. Examples of Criteria 1 tools which may require qualification include: design tools that generate source code (code generators); implementation tools that produce executable code representations; code representations or simulation tools (i.e. not actual); and binary translation tools such as cross-compilers or format generators.

Examples of Criteria 2 and 3 tools which may require qualification include: tools that automate code reviews and design reviews against standards; tools that generate test cases and/or procedures from requirements; tools that determine pass/fail status; tools that track and report structural coverage results; and tools which determine requirements coverage results.

Avionics code itself, compiler libraries, and Real-Time Operating Systems (RTOS's) are not considered tools since they form part of the actual executable software/hardware. They are verified by the design assurance process and require no tool qualification but rather full "flight software" certification per DO-178C.

Lifecycle For Qualified Tools

Quick question: can quality be built-in to a product after it's developed? Of course not: true product quality relies upon high-quality planning, implementation per plan, and assessment of implementation along with supporting processes. Just as DO-178C requires lifecycle processes for avionics software, DO-330 defines such a lifecycle for qualified tools as shown below in the following Figure. As shown in that following figure, the tool qualification lifecycle consists of three key activities: 1) Tool Planning, 2) Tool Development, and 3) Tool Verification; these activities must be performed sequentially, starting with Planning, then Development, and finally Verification. But continuously performed in the background during each of these activities are the corresponding Integral Processes of Tool Configuration Management, Tool Quality Assurance, and Tool Qualification Liaison.

vance.hilderman@afuzion.com
www.afuzion.com

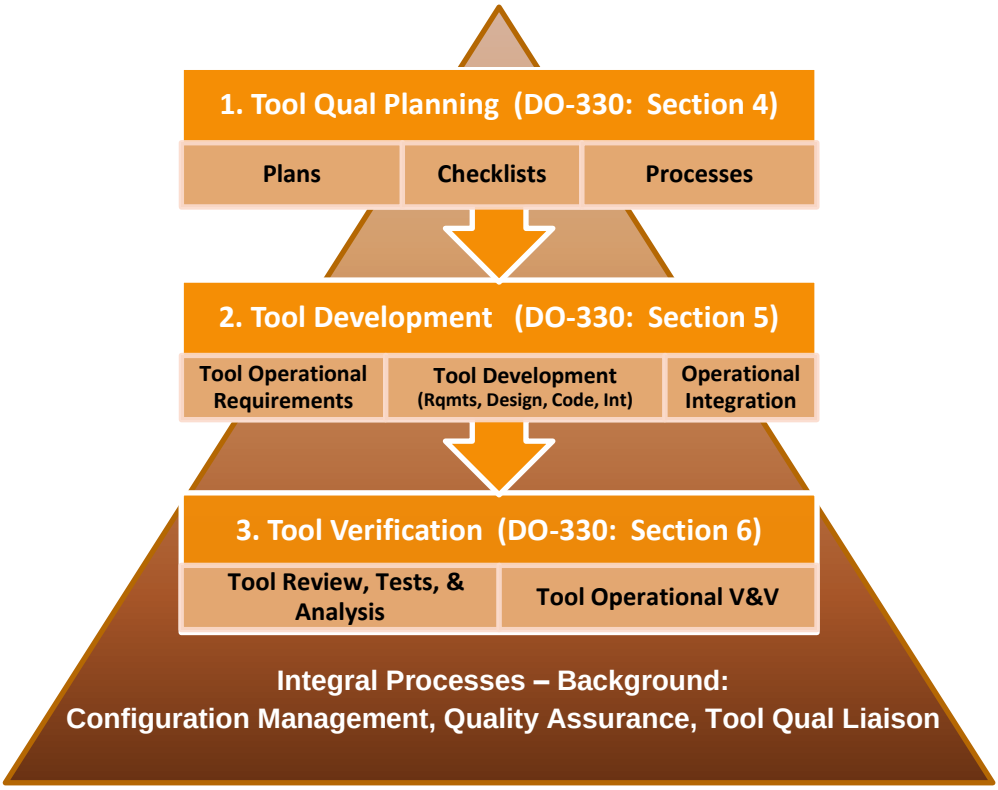


Figure 6: DO-330 Lifecycle for Qualified Tools

Designating The TQL: Tool Qualification Level

OK, you've identified all your tools, determined which must be qualified and by what Criteria, and are ready to perform the tool qualification starting with tool qualification planning. It is critical for the planner, or planning organization, to master the organization's or project life-cycle processes, and their mapping to DO-178C/DO-254 Objectives, especially if multiple tools contribute to an Objective. But what is actually required to perform the tool qualification itself? It depends entirely on the tool's TQL. Remember, the reason for the five different TQL's is due to the simple fact that the potential adverse effect of incorrect tool usage or output varies dramatically between TQL's.: TQL 1 tool problems are normally more threatening than TQL 5 tool problems. Therefore, TQL 1 tools require the most qualification rigor and artifacts, while TQL 5 tools require the least. If a specific tool is used as a TQL 3 tool, are you allowed to qualify it to a higher level, e.g. TQL 2 or TQL1? Yes, certainly; you're always allowed to do more work than required especially if you have unlimited budgets and schedule; but since you are intelligent, you would not do that unless you are reasonably sure you'll need to prove the higher TQL for a subsequent project. Are you

vance.hilderman@afuzion.com
www.afuzion.com

allowed to qualify to a lower level, e.g. TQL 4 or TQL 5? Absolutely not, as those lower levels (ascending TQL number means lower level) have fewer qualification objectives. Recommendation: avionics development is extremely expensive and time-consuming already; avoid extra work and qualify the tool to the minimum TQL required unless you are certain you'll re-use the tool on a different project requiring that higher TQL.

Planning the Tool Qualification.

Now, the TQL has been formalized, your certification authority has formally approved your plans or you are reasonably sure they will, so actual tool qualification can begin. The required tool qualification objectives that must be satisfied are detailed in the ten Annex A tables at the back of DO-330. These objectives depend upon the TQL, as summarized in the following figure: note that the required tool qualification objectives increase as the TQL advances from the least rigorous (TQL 5) to the most rigorous (TQL 1):

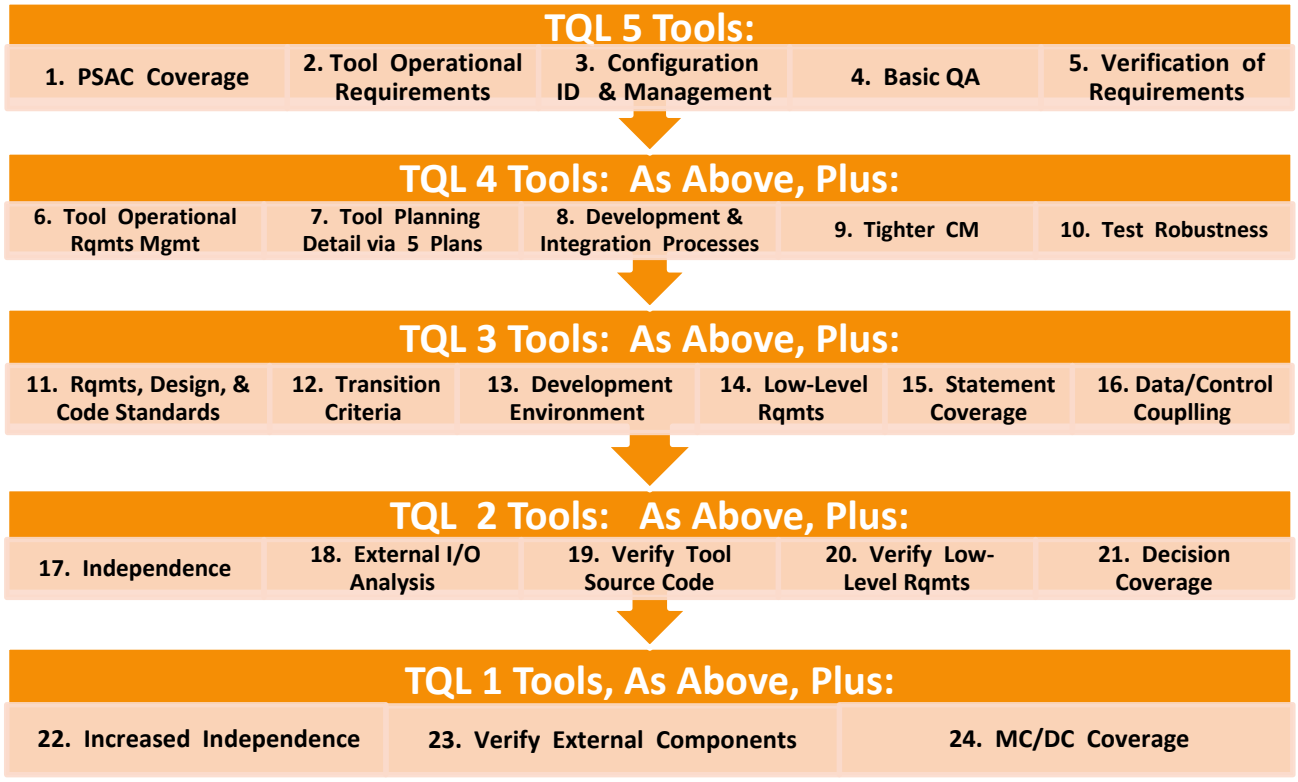


Figure 7: Key Tool Qualification Objectives & Data by TQL

The first step in planning for software tool qualification is to ensure the tool qualification is necessary, as previously explained. Presuming tool qualification is necessary, the figure above summarizes key qualification objectives and data. First, it is important to understand what objectives you will be required to meet based upon your tool's TQL. DO-330 lists each specific objective based upon TQL but the key objectives are cited in the figure above. Again, the key point to understand is that these tool qualification objectives are additive: as TQL rigor increases, from TQL 5 to TQL 1, additional objectives are required. Is

vance.hilderman@afuzion.com
www.afuzion.com

it possible that today's TQL 5 tool will be tomorrow's TQL 4 tool, or today's TQL 3 tool will be a TQL 2 tool tomorrow? Absolutely. If you even suspect that such an increase in your tool's TQL could be required, should you simply do the additional work today? Yes, if you have surplus budget and schedule; which means probably not as almost no aviation related project has such. If you think there is a reasonable probability your tool will need to be qualified to a higher TQL in the future, you should defer those additional objectives with the exception of "independence". Independence refers to the verification process (reviews, tests, analysis) and if verification was not performed with independence it would need to be done over for the higher TQL that required such. Recommendation: perform verification independently, even when not formally required; it may cost a little more for an independent engineer to gain familiarity with the technical artifacts, but the resultant independent verification will be of higher quality – just do it.

What Information & Data are Required for Tool Qualification?

The format and packaging of the tool qualification data needed to be submitted and made available for review is dependent on the type and TQL of tool being qualified. As expected, TQL 1 tools typically require the most data, whereas TQL 5 tools the least. The following Figure 8 summarizes typical tool qualification data by TQL and at what stage in the development cycle it is prepared. It should be noted that certification authorities are more concerned about the "quality" of tool qualification data than the "packaging". Thus there is much leeway to combine tool qualification data within few documents or even within corresponding application software development documents. While it might seem prudent to reduce the number of documents, be forewarned: good tools are leveraged on other projects or customized over time and therefore putting tool qualification data in separate documents actually simplifies re-use and re-qualification over time. And remember, the amount of data needed within each of the data items in the following figure is TQL dependent; therefore a thorough reading of all 138 pages of DO-330 should be performed to ensure the required data is included and also to avoid gathering and documenting data which is not required at less rigorous TQL's.

Why are some of the data items marked "M" (for "Maybe") in the following chart? At less-rigorous TQL's, certain data is either not required or can be placed in other documents. For example, data normally contained in a Tool Qualification Plan (TQP) for a TQL 5 tool can simply be included in the PSAC, thus negating the need for a separate TQP. Simple. One issue the planner will have to face is whether to produce one TQP per tool, or produce a single consolidated TQP for all tools. Some groups opt for the later on this latest project to show how all the Tools are integrated into the life-cycle process.

Many tool vendors do not provide a Tool Qualification Certificate with a completely executed Tool Qualification Plan and a complete set of Qualification artifacts. Instead, they provide a Tool Qualification Support Package which the organization has to take ownership of, tailor, as required, then execute to produce qualification artefacts. It is critical that the organization qualifying a tool get their hands on the Tool Qualification Support Package from the tool vendor as early as possible. Typically, a Tool Qualification Support Package contains: 1) a Tool Qualification Plan template; 2) the "default" Tool Operational Requirements; 3) a Tool Verification Plan; 4) Tool Qualification Procedures and Test Cases; 5) source code files or model files required to support the execution of the Test Cases; and 6) a pro-forma Tool Accomplishment Summary. In some cases, the Tool vendor will embed tailoring instructions in the template documents, others will provide a global description document. It is important to review and assess the quality of the tailoring instructions. Organizations will also need to assess the amount of Qualification

vance.hilderman@afuzion.com
www.afuzion.com

support they will require from the Tool Vendor to tailor the Tool Qualification Support Package, and sometimes, to execute it.

It should be noted the Tool Installation Report is not normally provided by the Tool Vendor and it will be critical that the organization get a good handle on this one. In some cases, it is preferable to produce a single consolidated TIR for all the tools.

vance.hilderman@afuzion.com
www.afuzion.com

DATA ITEM	STAGE			Applicable TQL				
	Liaison & Integral Processes	Tool Development	Tool Operation & Integration	TQL 1	TQL 2	TQL 3	TQL 4	TQL 5
Tool-Specific Information in PSAC	✓			✓	✓	✓	✓	✓
Tool Qualification Plan (TQP)	✓			✓	✓	✓	✓	M
Tool Development Plan	✓			✓	✓	✓	✓	M
Tool Verification Plan	✓			✓	✓	✓	✓	M
Tool Configuration Management Plan	✓			✓	✓	✓	✓	M
Tool Quality Assurance Plan	✓			✓	✓	✓	✓	M
Tool Standards (Requirements, Design, Code)	✓			✓	✓	✓	-	-
Tool Configuration Index	✓			✓	✓	✓	✓	M
Tool Lifecycle Configuration Index	✓			✓	✓	✓	✓	✓
Tool Problem Reports	✓			✓	✓	✓	✓	✓
Tool Configuration Management Records	✓			✓	✓	✓	✓	✓
Tool Quality Assurance Records	✓			✓	✓	✓	✓	✓
Tool Accomplishment Summary	✓			✓	✓	✓	✓	M
Tool Info in Software Accomplishment Summary	✓			✓	✓	✓	✓	✓
Tool Requirements		✓		✓	✓	✓	✓	✓
Tool Design		✓		✓	✓	✓	M	-
Tool Source Code		✓		✓	✓	✓	✓	✓
Tool Executable Code		✓		✓	✓	✓	✓	✓
Tool Verification Cases & Results		✓		✓	✓	✓	✓	✓
Trace Data		✓		✓	✓	✓	✓	✓
Tool Operational Requirements			✓	✓	✓	✓	✓	✓
Tool Installation Report			✓	✓	✓	✓	✓	✓
Tool Operational V&V Cases and Procedures			✓	✓	✓	✓	✓	✓
Tool Operational V&V Results			✓	✓	✓	✓	✓	✓

Figure 8: Typical Custom Tool Qualification Data by TQL

✓ = Data Item Typically Required

M = Data Item May Be Required Depending Upon Further Criteria

- = Data Item Typically Not Required

vance.hilderman@afuzion.com
www.afuzion.com

Tool Planning, Development, & Verification.

As outlined above, there are three key activities for producing qualified avionics tools:

1. Tool Qualification Planning
2. Tool Development
3. Tool Verification

Each of these three key activities of tool qualification are described below.

1. Tool Qualification Planning & Data.

It should be no surprise that the engineering of qualified avionics tools bears strong resemblance to the engineering of avionics: "plan it", "implement it per the plans", and then "verify it"; all while following the Integral Processes of Planning, then Development, with CM, QA and Liaison performed in the background. Tool qualification planning has numerous objectives including:

- Determine then define the tool's entire lifecycle and interrelationships between lifecycle processes; summarize such within a PSAC (and/or TQP).
- Identify the tool development and verification environments and related details, in advance.
- Specify applicable tool standards for requirements, design, and code; note these can be very similar to (or identical) with, similar standards for avionics software.
- Identify all applicable DO-330 objectives and define how each is to be accomplished.

The output of the tool qualification process will be applicable data including the following; :

- ✓ **Tool-Specific Information in PSAC**
- ✓ **Tool Qualification Plan (TQP)**
- ✓ **Tool Development Plan**
- ✓ **Tool Verification Plan**
- ✓ **Tool Configuration Management Plan**
- ✓ **Tool Quality Assurance Plan**
- ✓ **Tool Standards (Rqmts, Design, Code)**

2. Tool Development & Data.

AFTER the aforementioned tool qualification planning data are documented and reviewed, the tool implementation (or reverse engineering for pre-existing tools) is initiated to those plans. Applicable tool functional requirements, design, and code are developed in that order with transition and integration criteria (including traceability) affirmed and audited. Why are these important and required? Simple: to be qualified, a tool must undergo thorough verification including a minimum testing the tool's functionality versus the specified functionality in its requirements; tools with more rigorous TQL's will even have additional testing of robustness and structural coverage of the tool's source code. But testing alone can never by itself ensure high-quality. Like building a skyscraper, testing for earthquake survivability after the building is built is insufficient: such earthquake tolerances would have to be built into the building's architecture and considered throughout while building the foundation, walls, and floors. Same for avionics tools: quality provisions must be addressed throughout the development lifecycle. Hence DO-330's objectives for planning, processes, tool requirements/design/coding standards, integration, transition criteria, and traceability.

vance.hilderman@afuzion.com
www.afuzion.com

3. Tool Verification & Data.

Just as for certified aviation logic itself, the associated qualified tools used to develop or verify that certified logic require a defined verification process with specific objectives. Verification of qualified tools requires two sequential verification processes:

a) Tool verification process, followed by the:
b) Tool operational verification and validation process

"b" above is unique to tools: certification of avionics software (DO-178C) and hardware (DO-254) do not require operational verification and validation because the "operation" of such flight software and hardware is part of the system level requirements which are verified and validated at that that system level. However, unlike avionics hardware and software, tools do not have system level requirements. Therefore, tools qualified under DO-330 must have an additional process to ensure the tool can be operated as properly intended ("verification"), and that those corresponding tool operational requirements are correct ("validation").

Tool Requirements versus Tool Operational Requirements. Yes, qualified tools need two types of requirements: "Tool Requirements" which specify the tool's functionality, and "Tool Operational Requirements" which specify the tool's intended usage. Why two different types of requirements for tools? Remember, tools are commonly used by persons other than the developers of those tools. The tool developers' work, the tool itself, needs to be verified against the intended functionality of the tool which are expressed as Tool Requirements. The tool user's work needs to be verified against the intended usage of the tool which is expressed as Tool Operational Requirements. While there is a relationship between what a tool does versus how the tool is operated, the tool requirements are distinct from the tool operational requirements.

As a final word, it never hurts to ask for assistance when considering tools in the development process. SEEK GUIDANCE! Remember all tools must be considered, categorized, and possibly qualified. A good practice is to specify in the certification planning documents ALL tools which are to be used and whether or not qualification will be sought. Obtain agreement on tool categorization, tool qualification necessity, and tool qualification methods established early.

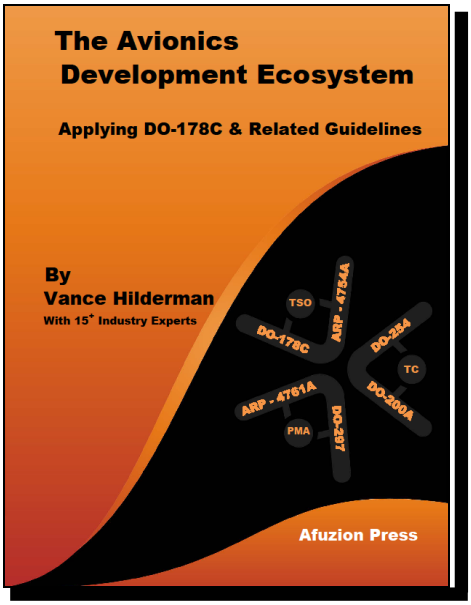
For more DO-178C Training information, see: **http://afuzion.com/training/**

For more DO-178C Gap Analysis information, see: **http://afuzion.com/gap-analysis/**

What is AFuzion? Fun One-Minute Video: https://www.youtube.com/watch?v=RMzLRzcahJE

vance.hilderman@afuzion.com
www.afuzion.com

For DO-178C & DO-254 specific details, procure the book “Avionics Certification: A Complete Guide To DO-178 & DO-254”, from major bookstores such as Amazon.com. (The author of this whitepaper is the primary author of that book.) Also, the new book “Avionics Development Ecosystem” by Vance Hilderman covers the big-picture view of avionics development from safety, to systems, and through all key regulatory and design aspects for modern avionics development. See the Afuzion website, www.afuzion.com, for advanced training modules relevant to DO-330 Tool Qualification and DO-178C for beginners and experts alike.



AFuzion’s Worldwide Onsite Engineering Footprint - *When Safety Is Critical ™:*

vance.hilderman@afuzion.com
www.afuzion.com