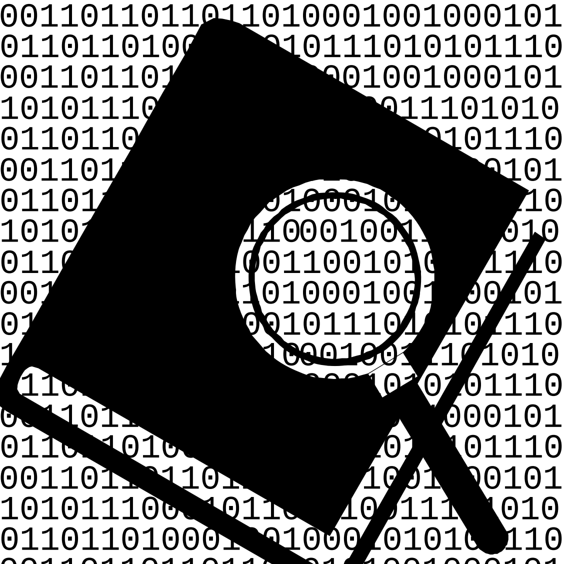
# Conducting Software Reviews Prior to Certification



## Job Aid

## AIRCRAFT CERTIFICATION SERVICE

Rev 1 – January 16, 2004

# Table of Contents

## ACRONYMS

| | |
|---|---|
| AIR | Aircraft Certification Service |
| API | Application Programming Interface |
| ASE | Aviation Safety Engineer |
| ASE-SW | Aviation Safety Engineer-Software |
| ASI | Aviation Safety Inspector |
| ASTC | Amended Supplemental Type Certificate |
| ATC | Amended Type Certificate |
| BSP | Board Support Package |
| CM | Configuration Management |
| COTS | Commercial-off-the-shelf |
| CPU | Central Processing Unit |
| CRC | Cyclic Redundancy Check |
| CSTA | Chief Scientific and Technical Advisor |
| DAR | Designated Airworthiness Representative |
| DER | Designated Engineering Representative |
| DMIR | Designated Manufacturing Inspection Representative |
| FAA | Federal Aviation Administration |
| FHA | Functional Hazard Assessment |
| HQ | Headquarters |
| HW/SW | Hardware/Software |
| I/O | Input and/or Output |
| ISR | Interrupt Service Routine |
| MC/DC | Modified Condition/Decision Coverage |
| MMU | Memory Management Unit |
| PDS | Previously Developed Software |
| POC | Point of Contact |
| PSAC | Plan for Software Aspects of Certification |
| PSSA | Preliminary System Safety Assessment |
| PI | Principal Inspector |
| PR | Problem Report |
| QA | Quality Assurance |
| RAM | Random Access Memory |
| RTOS | Real-Time Operating System |
| SCCB | Software Configuration Control Board |
| SCM | Software Configuration Management |
| SCMP | Software Configuration Management Plan |
| SDP | Software Development Plan |
| SGD | Software Grand Design |
| SQA | Software Quality Assurance |
| SQAP | Software Quality Assurance Plan |
| SQT | Software Qualification Test |
| SOI | Stage of Involvement |
| SOIs | Stages of Involvement |
| SOV | Shut-Off-Valve |
| SSA | System Safety Assessment |
| SVP | Software Verification Plan |
| SW | Software |

|  |  |
|---|---|
| STC | Supplemental Type Certificate |
| TC | Type Certificate |
| TS | Technical Specialist |
| TSOA | Technical Standard Order Authorization |

## *SUMMARY OF CHANGES*

The purpose of Revision 1 to the Software Review Job Aid is to implement references to current policy, address lessons learned since 1998, correct errors, and add special concerns (e.g, object-oriented technology, real-time operating systems, and reverse engineering). Significant changes to the Job Aid are summarized below.

| Change | Reason for the Change |
|---|---|
| Combined Sections 1-3 into a single "Part 1" | To condense up-front information and get to the main body of the document faster. |
| Moved Appendix A (example agendas, letters, and report) to Supplement 3. | To reduce the size of the Job Aid for those who physically carry it. |
| Moved Appendix B (the Summary of Compliances/ Findings/Objectives form) to Part 4 | The summary was moved to the main body to highlight the importance of summarizing the review results against the DO-178B objectives. |
| Added Supplements 1, 2, and 4. | To provide additional information for reviewers. The supplements provide useful information that may not be frequently needed. |
| Added the concept of documenting "compliances" | To encourage documentation of the good things, as well as the findings and observations during a review. |
| Added definitions for "actions" and "issues" | To provide an approach for complete documentation of the review activities. |
| Reworded the definitions of "finding" and "observation" | To be consistent with FAA Order 8110.49. |
| Added Appendix A | Documents an alternate approach for documenting review results. This approach has been effectively used on a number of reviews. |
| Added references and Appendix B | Inserted references to appropriate documents (for example, Order 8110.49), where possible. |
| Modified ASI role | Removed ASI involvement in software review process, since that direction was changed by FAA management. Added a QA/CM member to carry out this role. |
| Modified activities/questions in part 3 | To address lessons-learned over the past 5 years (e.g., from FAA real-time course, tutorials, object-oriented technology projects, handbooks, etc.). |
| Removed the column for check marks in the activities/questions tables in Part 3 | To reduce the "checklist" mentality. Reviewers may still wish to add columns to track activity, document issues, and document reviewed data. |
| Edited/reworded most existing questions in Part 3 | For clarity. In some cases the questions were also renumbered in order to improve the flow of the tables. |
| Added 1.1.2, 1.1.4, 1.1.5, 1.1.7, & 1.1.13-1.1.15 in Part 3 | To more thoroughly review plans upfront and to address interfaces between software personnel and systems, safety, and hardware personnel. This change led to some renumbering of activity 1.1 questions. |

| Change | Reason for the Change |
|---|---|
| Reworded question 1.1.6 in Part 3 | Removed reference to major/minor changes, since the Job Aid focuses on pre-certification rather than post-certification. |
| Modified & reordered questions 1.2.1 – 1.2.7 in Part 3 | For clarification and completeness. |
| Added 1.2.8 in Part 3 | To address reverse engineering concerns. |
| Modified questions 1.3.1-1.3.3 in Part 3 | To clarify the change process for plans and the tie to the safety assessment process. |
| Added questions 1.4.3-1.4.5 and 1.4.8 in Part 3 | To more completely evaluate the development process. |
| Modified/renumbered questions 1.4.6 & 1.4.9 in Part 3 | For completeness. |
| Added 1.5.5 in Part 3 | To address sub-tier suppliers. |
| Added 1.6.4, 1.6.5, 1.6.7-1.6.9, and 1.6.11 in Part 3 | To more thoroughly evaluate the SQA plan. |
| Added questions 1.7.3, 1.7.7 thru 1.7.14 in Part 3 | To more thoroughly evaluate the software verification plan. |
| Modified 1.7.5, 1.7.6 in Part 3 | For completeness. |
| Added 1.8.2-1.8.5 in Part 3 | For more complete evaluation of safety aspects. |
| Added 1.9.1-1.9.3 and 1.9.7 in Part 3 | For more complete evaluation of the software development standards. |
| Added 1.10 and questions in Part 3 | To address real-time software issues. |
| Added 1.11 and questions in Part 3 | To address object-oriented technology issues. |
| Added 2.1.14, 2.1.15, 2.1.16, 2.2.17, 2.3.12, & 2.4.9 in Part 3 | For more thorough evaluation of the development process. |
| Inserted activities/questions 2.5 and 2.6 in Part 3 | To address real-time software development issues. |
| Added questions 2.7.3 & 2.7.4 in Part 3 | To address configuration of tools and compiler options in development environment. |
| Added 2.12 in Part 3 | To address memory management. |
| Added 2.13 in Part 3 | To address software tools. |
| Added 2.14 in Part 3 | To address partitioning/protection. |
| Added 2.15 in Part 3 | To address real-time operating systems, when used. |
| Added 2.16 in Part 3 | To address object-oriented technology concerns. |
| Added 3.1 in Part 3 | To ensure SVP is being followed. |
| Added 3.2 in Part 3 | To address normal and robustness testing. |
| Added 3.3 – 3.5 in Part 3 | To implement review approach addressed in section 5 of the MC/DC tutorial [3]. |

| Change | Reason for the Change |
|---|---|
| Moved 2.5 to 3.6 in Part 3 | Integration activities are part of SOI #3 rather than SOI #2. |
| Added 3.6.2 in Part 3 | To address RTOS integration issues, if an RTOS is used. |
| Added 3.7 in Part 3 | To address data/control coupling. |
| Added 3.9.6 | To address low-level requirements coverage. |
| Added 3.12.5 in Part 3 | To address preservation of development environment. |
| Added 3.15 in Part 3 | To address tool qualification as part of SOI 3 (if tools are used). |
| Added 4.10 in Part 3 | To address the software conformity review. |

## *NOTES:*

## *PART 1 – OVERVIEW OF THE SOFTWARE REVIEW*

### *Purpose*

This Job Aid assists certification authorities, designees, and applicants in performing software reviews, as described in Chapter 2 of Federal Aviation Administration (FAA) Order 8110.49, *Software Approval Guidelines* [2]. The purpose of the software review is to assess whether or not the software developed for a project complies with the applicable objectives of RTCA/DO-178B, *Software Considerations in Airborne Systems and Equipment Certification* [1].

This Job Aid should be used as a reference tool during the software review process. It is **not** intended to be used as a checklist and is **not** all inclusive of all possible situations that need to be reviewed. Nor is the Job Aid intended to replace DO-178B—it should be used in conjunction with DO-178B. Likewise, this Job Aid may include questions that are not appropriate for the specific project being evaluated. Reviewers should keep in mind that each software project has some unique characteristics and should use the Job Aid as best fits the specific situation. This Job Aid only addresses the software review prior to certification/authorization for the following processes: Type Certificate (TC), Supplemental Type Certificate (STC), Amended Type Certificate (ATC), Amended STC (ASTC), or Technical Standard Order Authorization (TSOA).

For purposes of this Job Aid, *compliance, finding, observation, action,* and *issue* are defined as follows:

- A **compliance** is the satisfaction of a DO-178B objective.

- A **finding** is the identification of a failure to show compliance to one or more of the RTCA/DO-178B objectives.

- An **observation** is the identification of a potential software life cycle process improvement. An observation is not an RTCA/DO-178B compliance issue and does not need to be addressed before software approval.

- An **action** is an assignment to an organization or person with a date for completion to correct a finding, error, or deficiency identified when conducting a software review.

- An **issue** is a concern not specific to software compliance or process improvement but may be a safety, system, program management, organizational, or other concern that is detected during a software review.

The Job Aid will assist the software reviewer to do the following:

- ϖ Perform the reviews discussed in FAA Order 8110.49 [2].
- ϖ Perform tasks associated with conducting a software review.
- ϖ Document review compliances, findings, observations, actions, and issues.
- ϖ Link review compliances, findings, and observations to DO-178B objectives.

***Job Aid Layout***

This Job Aid addresses:

- ❖ Tasks to be performed before, during, and after a software review (Part 2).
- ❖ Activities and questions to be considered during a review (Part 3).
- ❖ An approach to correlate the findings, observations, and compliances to DO-178B objectives (Part 4).

Supplements to the Job Aid provide examples and supporting information to help reviewers. Supplements have been packaged separately from the Job Aid's main body to reduce the Job Aid size and to provide flexibility. The supplements are summarized as follows:

- ❖ Supplement 1 – Typical Roles and Responsibilities of the FAA's Software Team
- ❖ Supplement 2 – Typical Roles and Responsibilities of the Software Designee
- ❖ Supplement 3 – Examples (letters, agendas, and report)
- ❖ Supplement 4 – Optional Worksheets for Reviewers

***NOTE****: Other supplements may be added (as needed) and will be posted on the FAA's software web-site* [9].

***Key Players in the Software Review Process***

Below is a high-level description of the role of the key players in the software review process.

**Table 1. Key Players in the Software Review Process**

| Key Players | Primary Role in Software Review |
|---|---|
| ***Aviation Safety Engineer Software (ASE-SW)*** | • Responsible for the software approval on the project or system being reviewed.<br>• Typically serves as the software review team leader and is responsible for coordination, scheduling, and other review activities. If not designated as team leader, the ASE-SW reviews data and assists the team leader.<br>• Reviews the technical aspects of the software development process. |
| ***Aviation Safety Engineer (ASE)*** | • Works in propulsion, avionics/electrical systems, flight control systems, or mechanical systems with responsibility for approval of the overall system whose software is being reviewed.<br>• May not have software expertise, but is familiar with the system requirements and architecture; safety aspects; and system performance, operational, and functional expectations.<br>• May accompany the software review team to provide expertise on the systems aspects of the project and to review safety, operational, functional, performance, and system requirements; certification schedule performance concerns; and project issues.<br>• Needs to be kept informed of status on system, software, and hardware issues and their potential impact on system approval. |
| ***Aviation Safety Inspector (ASI)*** | • Principal inspector for the applicant or developer whose software is being evaluated.<br>• Performs conformity inspections on the software, per chapter 4 of Order 8110.49 [2]. |
| ***Flight Test Engineer/Pilot*** | • Evaluates the aircraft or the system installed on the aircraft by performing system demonstrations, simulations, and aircraft ground and flight tests.<br>• Evaluates the system performance on the aircraft and identifies any safety, operational, or performance concerns.<br>• Needs to be informed of status on system, hardware, and software issues and any potential limitations or impact on aircraft flight testing and certification schedule. |

| Key Players | Primary Role in Software Review |
|---|---|
| ***Chief Scientific & Technical Advisor (CSTA)*** | • Serves as a technical consultant on new, novel, or unique technologies, methods, or means of compliance that require expert review and input, as needed. |
| ***Technical Specialist (TS)*** | • Serves as a technical expert and a resource for the review team during the software review process.<br>• May serve as a link between the ASE-SW and the CSTA. |
| ***Project Manager*** | • Responsible for coordination, schedule, and oversight of the overall certification project. |
| ***Directorate Standards Staff*** | • Involved in situations that may require Directorate policy, issue papers, or special conditions for novel technology or methods.<br>• Provides technical expertise, when needed. |
| ***Headquarters (HQ) Software Personnel*** | • Involved in projects that may require changes or additions to national software policy.<br>• Provides technical expertise, as needed. |
| ***Designated Engineering Representative (DER) With Software Authorization*** | • Works on behalf of the FAA (Aircraft Certification Office) to review software compliance.<br>• Assists as part of the review team.<br>• Often performs review(s) prior to the FAA review to make preliminary compliance findings and to resolve any issues.  May be responsible for conducting review and providing review results to ASE-SW.<br>• May be responsible for recommending approval or approving the software aspects of the system or the system, if delegated. |
| ***Manufacturing Designees*** | • Works on behalf of the FAA (manufacturing organization) to perform software conformities, per chapter 4 of Order 8110.49 [2]. |

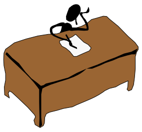
| Key Players | Primary Role in Software Review |
|---|---|
| ***Applicant*** | • Applies for Type Certificate, Supplemental Type Certificate, Amended Type Certificate, Amended Supplemental Type Certificate, or Technical Standard Order Authorization.<br>• Responsible for the compliance with applicable regulations, policy, special conditions, issue papers, and system and software policy. May or may not be the system or software developer.<br>• Responsible for oversight of system and/or software developer to ensure compliance, if applicable.<br>• Attends on-site software reviews. If the applicant is not the software developer, the applicant's oversight personnel, systems engineer, and/or other team members should be present at on-site reviews. |
| ***Software Developer*** | • May be the applicant or one of their system suppliers who is the developer of the system and/or the developer of the software under evaluation to be installed on an aircraft.<br>• Members of software developer team may include: software program manager, software development manager/lead, software verification manager/lead, software engineers, programmers, software quality assurance (SQA) personnel, software configuration management (SCM) personnel, etc. |

***NOTE:*** *Supplement 1 of this document provides the typical roles and responsibilities of the FAA software personnel. Supplement 2 provides an overview of the typical software engineering designee roles (the designee is typically a DER; however, designees in the Designated Alteration Station or other authorized organizations may perform similar activities).*

## *Review Types*

There are two types of reviews used to determine if an applicant's software development complies with the DO-178B objectives: (1) on-site review and (2) desk review. These reviews are performed either by the FAA or the FAA's designees, or both. Applicants or developers may also perform self-assessment reviews (for example, through their quality assurance organization). The table below provides a brief description of each type of review, when each type is appropriate, and the advantages and disadvantages of each. This Job Aid addresses both the on-site and desk reviews.

**Table 2. On-Site/Desk Review Summary**

| Type/Description | When Appropriate | Advantages/ Disadvantages |
|---|---|---|
| **On-Site Review:**<br>• Review appropriate stages of the software development process.<br>• Conducted by team at the software developer's facility. (Note: Applicant's designees and SQA should be present).<br> | • Highly critical systems (Levels A and B software).<br>• New system being developed.<br>• Integrated, complex avionics system with multiple functions.<br>• First-time applicants or first-time users of DO-178B.<br>• Applicant inexperienced in developing and/or overseeing software.<br>• Applicants with history of poor development processes.<br>• New or unique technology, software, methods, system or software architecture, safety proposals, partitioning concepts, etc.<br>• When a system demonstrates multiple problems during systems and flight testing.<br>• Major changes in the technology used or environment (e.g., personnel, tools, methods).<br>• At request of designee.<br>• FAA oversight of a designee is needed. | **Advantages:**<br>• Access to development personnel.<br>• More in-depth review.<br>• Higher confidence in safety aspects.<br>• Complete access to data.<br>• Helps perform designee performance evaluation.<br>**Disadvantages:**<br>• Budget and time considerations.<br>• May require travel.<br>• May impact developer's schedule. |
| **Desk Review:**<br>• Review appropriate software life cycle data.<br>• Conducted by individual or team at FAA or applicant's facility.<br>• Little or no involvement by the software developer.<br> | • Significant changes to previously approved software.<br>• Less critical systems (Levels C and D software).<br>• Experienced aviation companies with a good performance history.<br>• Confidence in designee is good. | **Advantages:**<br>• Not disruptive to the software developer's schedule and resources.<br>**Disadvantages:** (if conducted off-site)<br>• Many companies do not allow data to be viewed without their presence.<br>• Cannot ask direct questions of software developers.<br>• May require shipment of large amounts of data. |

The use of designees for software reviews is encouraged, when appropriate. Designees should be qualified and authorized for software. System designees are also encouraged to attend in order to contribute system domain expertise. The table below outlines when the delegation of a review is appropriate; additionally, some of the advantages and disadvantages of delegation are provided.

**Table 3. Delegation of Software Reviews**

| When Appropriate | Advantages and Disadvantages |
|---|---|
| • Designee has demonstrated good performance and has proper authorization.<br>• For all systems **without** unique software characteristics.<br>• When there are no issues that have policy implications (Directorate or Headquarters). | **Advantages:**<br>• Familiar with applicant or developer's processes and data organization.<br>• Usually has been or is located on-site and can monitor the processes throughout the development.<br>**Disadvantages:**<br>• May have company bias or have "ownership" perspective and not be impartial in assessing compliance.<br>• May have been promoted into management.<br>• May be pressured to "cut corners" when schedules are slipping. |

It is recommended that designees use the same type of review approach outlined in this Job Aid, plus any additional checks that they find necessary.

The software review may be used for the evaluation and subsequent approval of software data within the TC, ATC, STC, ASTC, and TSOA processes.

### *Stages of Involvement (SOI)*

DO-178B, Annex A includes ten tables, which outline the objectives that should be demonstrated by the applicant. The titles of the ten tables are provided below:

*A-1: Software Planning Process*
*A-2: Software Development Process*
*A-3: Verification of Outputs of Software Requirements Processes*
*A-4: Verification of Outputs of Software Design Process*
*A-5: Verification of Outputs of Software Coding and Integration Processes*
*A-6: Testing of Outputs of Integration Process*
*A-7: Verification of Verification Process Results*
*A-8: Software Configuration Management Process*
*A-9: Software Quality Assurance Process*
*A-10: Certification Liaison Process*

The purpose of the software review is to ensure compliance to the DO-178B objectives and other applicable software policy, guidance, and issue papers. To assess compliance, there are typically four Stages of FAA Involvement throughout the software life cycle of a project. The four SOI reviews are listed below and overviewed in Table 4:

(1) Planning Review;
(2) Development Review;
(3) Verification Review; and
(4) Final Review.

For each SOI review the following information is provided: a brief description of the SOI, required data, related DO-178B Annex Tables used as evaluation criteria, and related sections of this Job Aid.

Reviews may be combined or delegated to an authorized designee, as the project requires. Even if the FAA elects not to perform four reviews, it is strongly encouraged that designees perform on-site reviews and/or the applicant conducts internal compliance reviews using the approach outlined in this Job Aid.

For some projects more than four reviews may be warranted; for example, a large project with many sub-systems, an integrated modular avionics system, or a project experiencing multiple problems.

**Table 4. Overview of Stages of Involvement**

| SOI | Description | Data Reviewed | Related DO-178B Table | Related Job Aid Section |
|---|---|---|---|---|
| 1 | **Planning Review**<br>• Assure plans and standards meet DO-178B objectives and address other applicable software policy, guidance, and issue papers.<br>• Assure that the processes described in the applicant's plans meet the objectives of DO-178B and address other applicable software policy, guidance, and issue papers.<br>• Obtain agreement between FAA and applicant on the plans, standards, and proposed methods of compliance. | • Plan for Software Aspects of Certification (PSAC)<br>• Software Verification Plan (SVP)<br>• Software Development Plan (SDP)<br>• Software Configuration Management Plan (SCMP)<br>• Software Quality Assurance Plan (SQAP)<br>• Software Development Standards (Requirements, Design, and Coding)<br>• Safety assessment (preliminary system safety assessment (PSSA) or system safety assessment (SSA))<br>• Tool Qualification Plans, if applicable<br>• Other applicable company policy, procedures, and standards<br>• System requirements (may be preliminary) and interface specifications<br>• Description of any new technology or novel methods (typically described in the plans) | A-1, A-8, A-9, A-10 | Section 3.1 - Activities for SOI #1 |

| SOI | Description | Data Reviewed | Related DO-178B Table | Related Job Aid Section |
|---|---|---|---|---|
| 2 | **Development Review**<br>• Assess implementation of plans and standards for the software requirements, design, and code, and related verification, SQA, and SCM data.<br>• Assess and agree to plans and standards changes.<br>• Assess implementation of new technology and methods to ensure compliance to plans, standards, and agreements.<br>• Assure life cycle data satisfies DO-178B objectives and other applicable software policy, guidance, and issue papers. | • Software Development Standards (Requirements, Design, and Coding)<br>• Software Requirements Data<br>• Design Description<br>• Source Code<br>• Software Verification Results (as applied to Tables A-2 to A-5)<br>• Problem Reports<br>• Software Configuration Management Records<br>• Software Quality Assurance Records<br>• Tool Qualification Data, if applicable<br>• Resolution of previous review findings, if applicable | A-2, A-3, A-4, A-5, A-8, A-9, A-10 | Section 3.2 - Activities for SOI #2 |
| 3 | **Verification Review**<br>• Assess implementation of verification and test plans and procedures.<br>• Assess completion and compliance of all associated SCM and SQA tasks.<br>• Ensure software requirements are verified.<br>• Ensure robustness testing is planned and is being performed.<br>• Ensure analyses (including timing, memory, test coverage, structural coverage, | • Software Requirements Data<br>• Design Description<br>• Source Code<br>• Software Verification Cases and Procedures<br>• Software Verification Results (including review results, analyses results, and test results)<br>• Problem Reports<br>• Software Configuration Management Records<br>• Software Quality Assurance Records<br>• Resolution of previous | A-2, A-6, A-7, A-8, A-9, A-10 | Section 3.3 - Activities for SOI #3 |

| SOI | Description | Data Reviewed | Related DO-178B Table | Related Job Aid Section |
|---|---|---|---|---|
| | and data and control coupling) are being performed, as required by DO-178B.<br>• Ensure verification activities satisfy DO-178B objectives. | review(s) findings, if applicable | | |
| 4 | **Final Review**<br>• Assure final software product meets DO-178B objectives and is ready for certification.<br>• Address any open items. | • Software Conformity Review Results<br>• Software Life Cycle Environment Configuration Index<br>• Software Verification Results (final test, analyses, and review results)<br>• Software Configuration Index<br>• Problem Reports<br>• Software Accomplishment Summary<br>• Final resolution of all previous review findings and issues | All | Section 3.4 - Activities for SOI #4 |

### *Determining Level of Involvement*

As early as possible in the software project, the FAA (or designee, if delegated) should estimate the required level of involvement. The process for determining and documenting the FAA level of involvement is described in Chapter 3 of Order 8110.49 [2]. Early in the software project, the FAA (or designee) should evaluate such things as: the applicant/developer's experience; the history of the applicant/developer; the quantity/quality of designee support; the novelty/uniqueness of the system, technology, methods, or project; and the system criticality. Based on this early evaluation, the FAA will determine if the level of involvement is high, medium, or low. The level of FAA involvement will dictate the number of software reviews, the stages of involvement, and the nature of the review (i.e., desk or on-site).

For example, for a critical system being developed by a company who has never used DO-178B, the level of FAA involvement would likely be high and on-site software reviews would be performed at all Stages of Involvement. However, for a level D system being developed by an experienced company with a good history, the level of FAA involvement would be low and all reviews would likely be delegated to designees and be performed as desk reviews.

The FAA's involvement should be determined and documented as early as possible in the project.

### *The Review Team*

It is recommended that reviews be performed using a team of two to four people. A team can typically perform a higher quality review than an individual and can reduce the amount of time required to perform the review. The review team will typically divide responsibilities. Depending on the size of the team, there should be at least one "engineering (ENG) team member" and one "quality assurance/configuration management (QA/CM) team member." The ENG team member(s) will focus on development and/or verification data, while the QA/CM team member(s) considers the quality assurance and configuration management processes. Throughout this Job Aid the areas of responsibility for the ENG and QA/CM team members will be highlighted. In addition to the software engineers, there may be one or more non-software engineers as part of the team to oversee the systems, safety, and application aspects of the project. Additional team members may be a Chief Scientific and Technical Advisor, Technical Specialist, Directorate personnel, Headquarters personnel, or international certification authorities, as required.

## PART 2 – SOFTWARE REVIEW TASKS

### *Overview of Common Tasks*

Regardless of the SOI for the software review, the following tasks will be done for each review:

(1) Preparing for the review;

(2) Performing the review (referencing the appropriate Job Aid tables and documenting the review compliances, findings, observations, actions, and issues);

(3) Preparing and delivering an exit briefing of the review; and

(4) Conducting follow-up activities (e.g., prepare a report, assess if another review is required or if final compliance has been demonstrated).

The following pages give a detailed description of the four tasks performed by the review team in conducting a software review. The person(s) responsible for each task is listed next to the task. The “team leader” is the ASE-SW or designee responsible for leading the review. The “team” is typically comprised of ASE-SWs, designees, and others, as needed.

Once the SOI has been established, refer to the appropriate Activities/Questions Part 3 of this Job Aid. The Part 3 tables provide guidance as to what kinds of questions to ask to ensure that the DO-178B objectives are satisfied and other applicable software policy and agreements are complied with. If there is more than one SOI review, more than one set of tables will be referenced (e.g., if the review combines SOI #1 and SOI #2, tables for both SOIs would be used).

The Tasks and Activities/Questions outlined emphasize the on-site software review; however, the same types of activities are appropriate for a desk review. The desk review would require a different type of notification letter and access to the applicant’s personnel might be limited to telephone calls. However, the remainder of the activities are about the same. Places where the desk review differs from the on-site review are highlighted at the end of each task.

## TASK 1: Preparing for the Software Review

The purpose of this task is to assemble the review team, notify the applicant of the review, coordinate delivery of materials, and prepare all team members for the software review.

### STEP 1: COORDINATE WITH THE CERTIFICATION TEAM *(TEAM LEADER)*

1.1 Inform the project manager of the plans to conduct a software review and discuss all concerns (e.g., issue papers, project impact, team members, travel funds, foreign certification authority involvement).

1.2 Coordinate with and obtain necessary information from ASEs and Flight Test certification team members. (Note: It is often beneficial to have a systems ASE on the review team.)

1.3 Inform the Principal Inspector of review plans coordinate any concerns regarding conformity, if appropriate.

1.4 If the software review is to be performed at a software developer's facility located in another cognizant ACO's area of responsibility, contact the appropriate ACO engineer for coordination. The other ACO engineer may desire to be part of the review team in order to perform routine oversight roles.

1.5 Address any non-US certification concerns with the FAA certification team and the international certification team, if appropriate.

### STEP 2: ORGANIZE THE REVIEW TEAM *(TEAM LEADER)*

2.1 Determine the members of the review team, based on project needs.

- Team should consist of at least one software engineering team member (ENG) and one quality assurance/configuration management (QA/CM) team member.
- Designees assigned to the project should be involved as part of the review team.
- Aviation Safety Engineer (ASE), Principal Inspector (PI), Chief Scientific and Technical Advisor (CSTA), Technical Specialist (TS), Directorate software personnel, or Headquarters (HQ) software personnel may be part of the team, as needed.

2.2 Coordinate a date for the review with the applicant and team members.

## STEP 3: SEND A NOTIFICATION LETTER TO THE APPLICANT AT LEAST SIX WEEKS PRIOR TO THE REVIEW *(TEAM LEADER)*

3.1 The notification letter should inform the applicant of the: (1) purpose of the review; (2) proposed agenda; (3) data to be reviewed during the review; and (4) data to be sent to the review team members prior to the review. The applicant should be requested to send data to the review team members one month prior to the review start date.

- A sample notification letter and an agenda for each SOI may be found in Supplement 3, Section 1. The letter is about the same for each SOI; the agenda changes depending on the specific SOI.
- If the review combines Stages of Involvement, the contents of the agendas may be combined. For example, if a review was performed that combines the planning and development reviews (SOI #1 and SOI #2), the agenda, available data, and length of the meeting will need to combine the two sample agendas for SOI #1 and SOI #2 from Supplement 3, Section 1.

## STEP 4: COORDINATE WITH REVIEW TEAM MEMBERS *(TEAM LEADER)*

4.1 Assure that all review team members have copies of the Plan for Software Aspects of Certification (PSAC), Software Development Plan (SDP), Software Verification Plan (SVP), Software Configuration Management Plan (SCMP), Software Quality Assurance Plan (SQAP), Software Development Standards, and any other appropriate documentation at least two weeks prior to the review. (Note: These may not be the final copies of the plans, since they may change throughout the development process; however, they should be under configuration control.)

4.2 Assign responsibilities to team members:

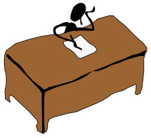
- All team members should review all plans and prepare a list of questions/concerns on those plans to clarify with the applicant at the review.
- The ENG team member(s) focuses on the software development processes, including the verification activities.
- The QA/CM team member(s) focuses on the QA and CM processes.
- If CSTA, TS, Directorate, or HQ personnel are to be involved in the review, communicate the area where their expertise is needed, so that they can prepare and perform any necessary research prior to the review.

4.3 All team members should review the activities/questions for the appropriate SOI review, as listed in Part 3 of this Job Aid.

## STEP 5: MEET WITH ALL TEAM MEMBERS PRIOR TO THE SOFTWARE REVIEW TO DISCUSS INDIVIDUAL RESPONSIBILITIES *(TEAM)*

5.1 This is typically a short meeting the evening or morning prior to the review. The purpose of this meeting is for all of the team members to be introduced, get a feel for any "issues" at the facility to be reviewed, get answers for any last minute questions relative to the review, and discuss any questions raised during preliminary review of software life cycle data.

> ***NOTE:*** *In some cases, a teleconference works for this pre-review discussion; particularly, when the team is spread over geographical distances.*



### Special Considerations for the Desk Review

ϖ When notifying the applicant of the review, make arrangements for when and where the data should be sent. Also, specify the number of copies needed.

ϖ Establish a Point of Contact (POC) at applicant's facility in case questions arise during the desk review.

ϖ Consider setting up a teleconference with the applicant at the end of each day or keep a log of questions to fax the applicant since in-person interviews won't be possible.

## TASK 2: Performing the Software Review and Documenting Compliances, Findings, and Observations

The purpose of this task is to conduct all activities necessary to complete the review; document compliances, findings, observations, actions, and issues; and determine next steps.

### STEP 1: CONDUCT ENTRY BRIEF WITH APPLICANT AND FAA TEAM *(TEAM LEADER)*

1.1 Introduce the review team members.

1.2 State the purpose of the review (summarize from the notification letter).

1.3 Review the agenda with the applicant and the appropriate personnel that need to be at each portion of the review.

1.4 Strive to create a good working partnership.

### STEP 2: PRESENT OVERVIEW OF APPLICANT'S SYSTEM, SOFTWARE, AND DEVELOPMENT PROCESS *(APPLICANT AND TEAM)*

2.1 Applicant presents program overview and designees' findings/observations/compliances from internal reviews.

2.2 During the applicant's presentation, the FAA review team focuses on those issues/processes compatible with their expertise. For example:

- ENG team member focuses on the software development process (and related verification activities) and technical issues.

- QA/CM team member focuses on implementation of SQA and SCM processes.

- If a system engineer is present, they should focus on the interfaces between the software engineering (development) processes and the systems engineering processes (including the system safety assessment process) and system verification and validation.

- The depth of the applicant's overview will depend on the type of review and whether or not the review team is familiar with the system or process (e.g., if this is a second review with the same team, the presentation may only be a memory jogger).

2.3 The review team assures that the applicant's presentation provides adequate information to give insight into the developer's processes and procedures.

- The review team members ask as many questions as needed to understand the software being reviewed.

2.4 The applicant's overview is generally limited in time to allow for adequate time to review data. The sample agendas, found in Supplement 3, Section 1, give a rough estimate of how much time should be allowed for the applicant's overview. (Note: The team leader should work to keep the review on schedule.)

2.5 It is recommended that the developer, applicant, or designee conduct a "self-assessment" of the project using the appropriate matrices for the type of review; and that they reveal any findings, observations, anomalies, potential issues, policy interpretations, or questions during the applicant overview session. This will enable the review team and applicant/developer to focus on coming to resolution of those items before the review is completed.

## STEP 3: REVIEW APPLICANT'S SOFTWARE DATA AND INTERVIEW THE APPROPRIATE PERSONNEL *(TEAM)*

3.1 Part 3 of this Job Aid provides a summary of activities to be performed and questions to be answered during a review. The particular SOI being reviewed dictates which table is referenced. If the review is a combination of SOIs, multiple tables should be used.

3.2 The activities/questions outlined in Part 3 are a guide. There may need to be more or less activities/questions, depending on the nature of the project. Also, the sequence of activities/questions is flexible. The goal of the Job Aid is to provide a standardized approach for getting started. This includes sufficient guidance to cover many situations. Each software development project will have unique aspects that are impossible to capture in a "standardized" Job Aid. If there is anything that is not understood during the review, the review team, the applicant, and the developer should discuss it further, and attempt to come to an agreement during the review.

## STEP 4: DOCUMENT THE SOFTWARE REVIEW COMPLIANCES, FINDINGS, OBSERVATIONS, ACTIONS, AND ISSUES *(TEAM)*

4.1 Document the compliances, findings, observations, actions, and issues in detail. (Compliances, findings, observations, actions, and issues are defined in Part 1 of this Job Aid in the "Purpose" section). Throughout the review, each team member should take notes on what documents (number, revision level, date) were reviewed, what thread traces were conducted, areas of concern (issues), non-compliance to DO-178B objectives (findings), discussions with the developer and applicant, and any other concerns or issues.

The review compliances, findings, observations, actions, and issues are included in the exit briefing and review report and must be documented in detail. The compliances, findings, and observations, should be documented in enough detail that they could be found again by another reviewer (i.e., reviews must be repeatable). Issues and actions outside the scope of the software aspects should be communicated to project management and appropriate specialists (systems, safety, flight test, etc.) for resolution.

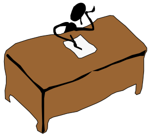
## STEP 5: MEET AT THE END OF EACH DAY TO ASSESS PROGRESS, SUMMARIZE AND DOCUMENT COMPLIANCES, FINDINGS, OBSERVATIONS, ACTIONS, AND ISSUES, AND PLAN FOR THE NEXT DAY *(TEAM)*

5.1 The team leader should keep a list of compliances, findings, observations, actions, and issues based on input from team members. [Note: Team members are encouraged to provide their list of items to the team leader in a manner that minimizes the team leader’s workload.]

5.2 Team members should discuss any concerns, questions, etc. and come to a common understanding and agreement on how to proceed.

5.3 The team should plan activities for the next day.

**Note:** The “end-of-the-day” meeting should be held in privacy to discuss issues openly (i.e., this meeting typically excludes the development team). Designees should be involved in the “end-of-day” meetings. During these meetings you may consider starting the DO-178B Compliances/Findings/Observations table, if time permits. (This will save time required for follow-up activities.)



**Special Considerations for the Desk Review**

ϖ The entry brief would only be a discussion among the review team members since the review is not on-site.

ϖ The applicant would not present the program overview—this information will be obtained by reading documents.

ϖ It will not be possible to interview personnel or witness development activities; however, you may want to keep a list of questions to fax to the applicant or discuss over the telephone.

## TASK 3: Preparing and Conducting Exit Briefing

The purpose of this task is to summarize compliances, findings, observations, actions, and issues and to share them with the applicant. The exit briefing addresses all issues related to DO-178B compliance, other related issues, and next steps.

### STEP 1: PREPARE FOR THE BRIEFING BY HAVING A REVIEW TEAM MEETING *(TEAM)*

1.1 Discuss compliances, findings, observations, actions, and issues; and agree on recommended action.

1.2 Determine and agree on how to debrief the applicant.

1.3 Summarize individual team member compliances, findings, and observations in sufficient detail for incorporation into the report.

1.4 Prepare summary of issues based on "end-of-day" meetings.

1.5 Organize presentation order for the exit briefing and agree on who will present. (In most cases, the team leader will brief all compliances, findings, observations, actions, and issues. However, in some cases individual team members will brief their own.)

### STEP 2: PRESENT EXIT BRIEFING *(TEAM)*

2.1 Provide introduction to exit briefing. (Team Leader)

v Thank the applicant for the cooperation extended.

v Be attentive to the applicant's organizational environment and its concerns and issues.

v Present information in an objective, positive manner.

v Give an overview of the review's purpose and briefing content.

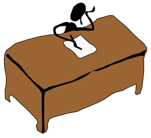
v Compliment and thank designees, applicant, and software developer, as appropriate.

2.2 Present compliances, findings, and observations in relationship to DO-178B objectives. (Team and Team Leader)

2.3 Summarize all issues that may be relevant to certification and compliance. (Team Leader)

2.4 Summarize any actions that must be taken (e.g., another review may be required to determine progress or the designees may need to perform an in-house review). (Team Leader)

2.5 Inform applicant that a report will be prepared by the FAA and sent with a letter to summarize and document findings, observations, compliances, issues, and actions. (Team Leader)



### Special Considerations for the Desk Review

ϖ A briefing or summary of the review should still be prepared; however, the delivery may be by teleconference or written report.

# TASK 4: Conducting Follow-up Activities

The purpose of this task is to perform follow-up activities after the review. This may include teleconferences; additional reviews with the applicant; and/or applicant oversight of the resolution of findings, issues, and actions. A report of the software review compliances, findings, observations, actions, and issues is prepared, coordinated, and sent to the applicant with an explanation of their role in oversight of resolution of the review results, and any future FAA or designee activity planned.

## STEP 1: PREPARE REPORT *(TEAM LEADER)*

1.1 Based on the team members' inputs at the review, prepare a report on the results of the review:

- Summarize all data reviewed, activities witnessed, and personnel interviewed.
- Include a description of the difference between a *finding, observation,* and *issue* (i.e., summarize the definitions in the Part 1 "Purpose" section of this Job Aid).
- Summarize compliances, findings, observations, actions, and issues in the report. It is recommended that the compliances, findings, and observations be documented with reference to the specific DO-178B objective(s) (see Part 4 of this Job Aid). (Note: An electronic copy of this Job Aid is provided on-line for use in the review process. You may use the electronic tables from Part 4 as the template for the compliances, findings, and observations to be included in the review report. The actions and issues are typically recorded in a separate section of the report.)

1.2 Coordinate the report with review team members.

- Complete no later than one week after the review so that a timely response is provided to applicants to ensure timely action and resolution.
- Work portions of the report at "end-of-day" meetings.
- Prepare report depth and length as appropriate to the situation.

## STEP 2: COORDINATE PRELIMINARY FINDINGS AND OBSERVATIONS WITH APPLICANT *(TEAM)*

2.1 Provide a draft report to the review team members to get clarification of any ambiguous or missing information, and get team agreement on the report contents.

2.2 After the review team agrees on the report contents, provide a draft copy to the applicant and developer and request a response to issues on which they do not agree.

2.3 Reconcile discrepancies with the applicant (via teleconference, e-mail, etc.).

2.4 Modify the report, as appropriate.

## STEP 3: DETERMINE FUTURE ACTIVITIES *(TEAM LEADER)*

3.1 Determine if there will be a need for another FAA or designee review.

3.2 Identify activities that the applicant must address to resolve deficiencies and show compliance.

3.3 Determine if issue papers need to be prepared and submitted to the applicant.

## STEP 4: PREPARE A LETTER TO THE APPLICANT AND SUBMIT FINAL REPORT *(TEAM LEADER)*

4.1 Address certification and non-compliance issues in the letter.

4.2 Attach the review report and summarize the findings and issues.

4.3 Describe future steps (actions) required by the applicant, designee, and/or FAA.

4.4 Summarize future activities, expectations, and plans (e.g., is a follow-up review required? should designees perform additional reviews?)

- Place a copy of the letter in the project file.

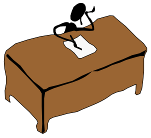
Note: Supplement 3, Section 2 contains a sample follow-up letter and report.

## STEP 5: COORDINATE WITH CERTIFICATION TEAM *(TEAM)*

5.1 Discuss appropriate issues with the Project Manager, system engineer, other engineers, flight test personnel, human factors personnel, Principal Inspector, other managers, etc. for the certification project.

5.2 Generate and provide issue papers, special conditions, etc. with the assistance of CSTAs, TSs, Directorate personnel, and headquarters personnel, as needed.

5.3 Address any foreign certification issues, such as the compliance with Certification Review Items (CRI), reporting on review activities, joint validation issues, etc.



**Special Considerations for the Desk Review**

- Since the applicant was not present during the review and did not have a chance to address concerns, it is important to set up a teleconference or meeting to address all issues and questions prior to the submittal of the report.

## NOTES:

# *PART 3 - ACTIVITIES FOR EACH STAGE OF INVOLVEMENT*

This part highlights the activities and questions to be performed by the software review team at each SOI. These activities/questions may vary slightly for a desk review (e.g., you won't be able to interview developers or observe activities for the desk review).

As emphasized before, these activities/questions are to be used as a guide. Different systems, projects, technology, methods, or situations may require the deletion or addition of activities/questions. An electronic copy of this Job Aid is available on-line so that you may tailor the activities/questions for your specific needs and situation. For example, you may desire to combine two SOI reviews, add or delete questions, and add or delete activities.

Each SOI review has a table summarizing the activities and questions that support that activity, with a correlation to the relevant DO-178B objectives. For novel or unique technology or methods, other questions and activities may need to be added. Review teams typically divide the activities and questions among team members, as appropriate.

It should be noted that the column for "checking" the items and indicating team responsibilities was removed from this version of the Job Aid to allow for more flexibility. Users of this Job Aid are encouraged to add columns that will help them with the review effort. For example, some reviewers may desire to add a column to track their activity and make notes. Some reviewers might also add a column to record the means of compliance and data reviewed (with data numbers and revisions).

## *NOTES:*

## 3.1 ACTIVITIES FOR STAGE OF INVOLVEMENT #1 – PLANNING REVIEW

***Purpose***

- To assess the interfaces with the system development process, hardware development process, and system safety assessment process; to review the system architecture; to assess the assigned software levels; and to determine the appropriateness of any system and hardware safety features and safety monitoring software and protection mechanisms for supporting systems reliability, integrity, safety, functionality, performance, and operability requirements (safety objectives).
- To assess the applicant's plans and standards in relationship to identified software level, safety features, and safety-related software requirements.
- To ensure that plans and standards meet objectives in DO-178B Annex A tables A-1, A-8, A-9, A-10, and that the software will comply with other applicable software policy and guidance.
- To assess that when the applicant follows their plans, they will meet all applicable objectives of DO-178B (Tables A-1 to A-10) and other applicable software policy or guidance.

***When Review Occurs***

Shortly after completion of the software planning process, or any other necessary point. Plans and standards may not be fully completed; however, they should be fairly mature prior to the review and under configuration control.

***Data Reviewed Prior to Review***

Plan for Software Aspects of Certification (PSAC), Software Verification Plan (SVP), Software Development Plan (SDP), Software Configuration Management Plan (SCMP), Software Quality Assurance Plan (SQAP), and Software Development Standards (Requirements, Design, and Coding).

***Data Reviewed at Review***

PSAC, SVP, SDP, SCMP, SQAP, findings/observations/issues from pre-review activities by designees/applicants/developers (if applicable), software development standards, safety assessment, system architecture, software level justification, safety features, tool qualification plans (if applicable), company policy and work instructions referenced in the plans and standards, and other data deemed necessary.

***Agenda***

See Supplement 3, Section 1.

***Number of Days Required***

1-3 days

### *Evaluation Activities and Questions*

The table below provides typical activities and questions for SOI #1.

### *Instructions*

- There are eleven major evaluation activities for Stage of Involvement #1: Planning Review.
- Review the questions for each activity in relationship to its corresponding DO-178B objective and software level.

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| **1.1** | **Review all plans (PSAC, SCMP, SQAP, SDP, SVP, software tool qualification plans, etc.) and standards. Based on your review of all the plans, consider the following questions:** | |
| 1.1.1 | Has the planning data been signed and put under CM (CC1 or CC2 as appropriate for the software level)? Verify there is objective evidence of coordination (e.g., authorized signatures) from all organizations controlled and affected by the software plans and standards. | • A-1, #1-7 |
| 1.1.2 | Have the plans and standards been reviewed and approved by a software designee (as authorized)? | • A-1, #1-7 |
| 1.1.3 | Are plans and standards cited complete, clear, and consistent? | • A-1, #1,7 |
| 1.1.4 | Were reviews of the plans and standards conducted and review results retained? And, were review deficiencies corrected? (See section 4.6 of DO-178B.) | • A-1, #6,7 |
| 1.1.5 | Do the plans and standards comply with the content as specified in DO-178B section 11 (i.e., sections 11.1 through 11.8)? Note: The plans and standards are not required to be packaged as identified in 11.1 through 11.8; however, the items specified in 11.1 through 11.8 should be documented somewhere in the plans and standards. | • A-1, #1-7 |
| 1.1.6 | Do the plans and standards address the software change process and procedures for the airborne software and tools (if tools are used)? | • A-1, #1,2 |
| 1.1.7 | Are all software tools identified in the plans and is rationale included for why each does or does not need to be qualified? | • A-1, #4 |
| 1.1.8 | Are the inputs, activities, transition criteria, and outputs specified for each process? | • A-1, #1 |
| 1.1.9 | Are the development and verification life cycle activities defined in sufficient detail (reference DO-178B sections 11.1-11.3) to satisfy section 4.2 of DO-178B? | • A-1, #1-7 |
| 1.1.10 | Do the plans and standards meet the DO-178B planning objectives in Table A-1? (i.e., Is each plan and standard internally consistent? Are the plans and standards consistent with each other? Is the software life cycle defined? Are the transition criteria defined?) | • A-1, #7 |
| 1.1.11 | If the plans and standards are followed, would this ensure that all applicable DO-178B objectives in Tables A-2 through A-10 are met? (I.e., do the plans and standards address how each of the applicable DO-178B objectives will be satisfied?) | • A-2 to A-10<br>• (all objectives) |
| 1.1.12 | Are the plans sufficiently clear and detailed to allow the development and verification engineers to follow them? | • A-1, #1-7 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.1.13 | Are the interfaces and communication channels with the system development processes addressed in the plans and well defined? Evaluate processes for flow down of system requirements (functional, performance, operational, safety-related, system architecture safety features, databus usage, I/O device usage) and for clarifying ambiguous system requirements. Determine if the software verification process will be claiming “formal test credit” for testing conducted by the system verification and validation processes, what kind of credit, and how much credit. Determine how and the environment that the developer will be performing worst-case timing analyses, memory usage analysis, etc. | • A-1, #1, 7 |
| 1.1.14 | Are the interfaces and communication channels with the system safety assessment process addressed in the plans and well defined? Evaluate the flow down of safety-related requirements (software partitioning, safety monitoring software, built-in test, etc.) and safety objectives from the system safety assessment (SSA) process to the software processes, and the feedback from the software requirements and design processes of derived requirements to the SSA process for evaluation. | • A-1, #1, 7 |
| 1.1.15 | Are the interfaces and communication channels with the complex electronic hardware development and procurement processes addressed in the plans and well defined? Evaluate the processes for documenting and communicating software dependencies and interactions with the hardware and its development/ procurement processes, and for hardware dependencies on the software and its development and verification processes. Determine the process used to convey hardware safety features (e.g., watchdog timers, built-in test, memory management unit, I/O device, processor features (cache, registers, priorities, schedulers, supervisor/user modes, etc.)) and hardware changes to the software processes, and how the software processes will address them. | • A-1, #1, 7 |
| **1.2** | **Determine if additional considerations defined in Section 12 of DO-178B have been documented and addressed in the plans. Consider the following questions:** | |
| 1.2.1 | Are such items as previously developed software, COTS software, user-modifiable software, field-loadable software, option-selectable software, multiple-version dissimilar software, product service history, alternative methods of compliance, etc. identified and addressed in the plans? | • A-1, #4 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.2.2 | If the developer plans on using previously developed software (PDS) for the current development, consider the following questions:<br>• Have any issues regarding modification to legacy systems or reuse of previously developed software been addressed in the plans? (Reference chapter 10 of Order 8110.49 for use of DO-178B for legacy systems.)<br>• If PDS from legacy systems is intended to be used, does the service history of the system support reuse of its software?<br>• Are there any airworthiness directives, service bulletins, in-work National Transportation Safety Board safety recommendations, or unresolved problem reports with safety, functional, performance, operational or maintenance issues for the legacy system or any proposed PDS?<br>• Does the PDS have a satisfactory service history?<br>• Does the developer intend to make any modifications to PDS? Are plans and processes defined for managing, controlling, and verifying those changes (in compliance with DO-178B section 12.1, 12.1.1-12.1.6)?<br>• Is the PDS or legacy system software used in an identical manner and executed on the same hardware and in the same environment as its previous uses? | • A-1, #4 |
| 1.2.3 | Verify that software tools are identified and explained in the plans. Consider the following questions:<br>• Do the plans provide rationale for why tools do or do not need to be qualified? (I.e., Does the use of tools result in the elimination, reduction, or automation of processes or activities found in DO-178B? Is the output of the tool verified by manual (review) or other means (another tool or activity)?)<br>• Is service history claimed for the use of any tool? If so, has the tool changed or is it being used in the same way as previously used? Does the documented tool service history support the intended use for the current development? | • A-1, #3, 4 |
| 1.2.4 | Are tools to be qualified supported with a tool qualification plan (either in the PSAC or in a separate document)? Verify that tools are properly categorized into development, configuration management, or verification tools. Verify that the plan for qualification of tools is documented and adequate for the specified tool use.<br>Note: Section 12.2 of DO-178B and Chapter 9 of Order 8110.49 provide specific guidelines regarding software tool qualification. | • A-1, #3, 4 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.2.5 | Are there unique additional considerations associated with the project (unique alternative means or methods of compliance, unique approaches to development, verification, SCM, SQA), proposals that don't comply with FAA national or Directorate policy or issue papers, etc.? If so:<br>• Have national or directorate software and systems personnel reviewed and approved any unique additional considerations?<br>• Is rationale for acceptance or rejection of developer/applicant's proposals well documented?<br>• Did the proposal and agreement/rejection result in new policy, an issue paper, or other means of documenting the decision?<br>• Did the decision result in any precedence for future system or software approvals? | • A-1, #4 |
| 1.2.6 | Are issue papers or national/directorate policy required for any of the additional considerations? If so, do the plans and standards address how compliance with the issue papers and/or national/directorate policy will be achieved? | • A-1, #4 |
| 1.2.7 | Have all foreign certification issues (e.g., certification review items, certification action items, review findings/issues/actions, etc.) been addressed (if a joint certification or validation project)? Additionally, do the plans and standards address how the foreign certification authority issues and concerns will be addressed? | • A-1, #4 |
| 1.2.8 | Is reverse engineering being planned for any PDS? If so:<br>• Has the rationale for reverse engineering been documented and adequately justified (typically in the "additional considerations" section of the PSAC)?<br>• Has the reverse engineering effort been planned (in the PSAC and other plans)?<br>• Are processes and procedures well defined?<br>• Is the reverse engineering life cycle documented?<br>• Does the reverse engineering approach meet DO-178B objectives?<br>• How will the high-level requirements be created? Is the plan adequate?<br>• Do transition criteria and traceability exist? Are they adequate?<br><br>**Note:** See CAST-18 [6] for specific certification concerns regarding reverse engineering. In some cases, an issue paper may be required, if the plans do not address the DO-178B objectives compliance. | • A-1, #4 |
| **1.3** | **Review PSAC and consider the following questions:** | |
| 1.3.1 | Does the PSAC adequately address the proposed contents described in DO-178B, Section 11.1? If not, where are the contents documented? [Note: If it is documented in another data item, that data item will need to be evaluated before the PSAC can be approved. Also, that data item should be configuration controlled to the same CC level as the PSAC.] | • A-1, #1-7<br>• A-10, #2 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.3.2 | Does the PSAC address the following questions regarding changes to plans and standards:<br>• Is a process in place to address changes to plans and standards that may occur throughout the development process?<br>• Are there plans and processes to address any deviations to plans and standards?<br>• Do the deviations require justification and rationale for why they are acceptable for this project?<br>• Will applicable aspects of the software plans, standards and procedures be conveyed to any sub-tier suppliers of components of the system and subcontractors to ensure their compliance to the approved plans, standards and procedures?<br>• Are the plans and standards under change control? | • A-1, #1,2 |
| 1.3.3 | Does the system safety assessment adequately support the software level proposed in the PSAC? If the software level is lower than what the system safety assessment suggests, is there adequate justification (e.g., through system architecture, safety feature, redundancy, fail safe design techniques, partitioning)? [Note: This determination will likely require input from the systems engineer.] | • A-1, #1-4 |
| **1.4** | **Review SDP and consider the following questions:** | |
| 1.4.1 | If the SDP is followed, will the DO-178B objectives defined in Table A-2 be met? (Note: Some objectives in Tables A-3, A-4, and A-5 may also be addressed in the SDP.) | • A-2, #1-7 |
| 1.4.2 | Does the SDP adequately address the proposed contents described in DO-178B, Section 11.2? If not, are the contents included in another plan? | • A-1, #1-4 |
| 1.4.3 | Are the software development processes defined in sufficient detail to ensure proper implementation of the software life cycle processes and model proposed for the project? Are transition criteria clear and enforceable? | • A-1, #1-4 |
| 1.4.4 | Will applicable aspects of the SDP, development environment, and procedures be conveyed to any sub-tier suppliers of components of the system and subcontractors to ensure their compliance to the approved plans, standards, and procedures? | • A-1, #1-4 |
| 1.4.5 | Are software development standards referenced in the SDP? | • A-1, #1,3,5 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.4.6 | Has the software development environment been adequately defined (e.g., documentation tools, requirements definition and capture tools, traceability tools, design tools (including architecture, derived requirements and low-level requirements definition and capture tools), coding tools (including code templates, code editors, compiler options and optimizations to be used), integration tools (including linkage editors and loaders, linking and loading procedures and tools), development host computer environment, tools to ensure protection of baselined software life cycle data such as configuration management and control tools, access privileges, etc.)? Additionally:<br>• Are tool users' guides, restrictions, and limitations available and known by the software developers using them?<br>• Do any of the tools support enforcement of the software standards, transition criteria, data baselining and approval process, etc.? For example: (1) does the code editor tool or compiler enforce any coding rules, restrictions, or limitations? (2) does the document control (CM system) enforce access privileges to data and ensure no unauthorized changes to baselined data? | • A-1, #3 |
| 1.4.7 | What kind of compiler is being proposed? Does the applicant have experience with the compiler? Have the compiler options and optimization been identified? (Note: Changes to compiler options and optimization may invalidate previous tests and coverage analysis.) | • A-1, #3 |
| 1.4.8 | Is the programming language and operating system specified and will they meet the objectives of DO-178B? (Note: Some language and operating system choices may produce non-deterministic results and therefore may not meet the objectives of DO-178B.) | • A-1, #3 |
| 1.4.9 | Is the "host" software development environment similar to or identical to the target environment? Consider the following questions:<br>• Will a different compiler, linker, or loader be used when integrating the software into the target environment than will be used in the "host" environment?<br>• Have all differences been identified, analyzed and justified? (For example, will the results of verification (analyses and testing) conducted on software in the host environment still be valid for the software in the target environment?)<br>• Will timing, memory and resource management analyses and testing have to be repeated for the target environment? | • A-1, #3 |
| **1.5** | **Review the SCM plan and consider the following questions:** | |
| 1.5.1 | If the SCM plan is followed, will the DO-178B objectives defined in Table A-8 be satisfied? | • A-8, #1-6 |
| 1.5.2 | Are the SCM processes as defined in Section 7.0 of DO-178B described in sufficient detail (see DO-178B Sections 11.4) to satisfy Section 4.2? | • A-8, #1-6 |
| 1.5.3 | Does the SCM plan adequately address the proposed contents described in DO-178B, Section 11.4? If not, are the contents included in another plan? | • A-8, #1-6 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.5.4 | Does the SCM plan provide for the following items?<br>• Configuration identification of software life cycle data.<br>• Baselining of all configuration control 1 (CC1) data.<br>• Problem reporting, change control, and configuration status accounting.<br>• Archival, retrieval, and release.<br>• Data retention provisions supporting airworthiness requirements.<br>• Software load control and part numbering to include any additional considerations required for electronic part numbering.<br>• Configuration management of the software life cycle development environment includes tools.<br>• All DO-178B life cycle data to be maintained consistently with the configuration control category associated with the software level. | • A-8, #1-6 |
| 1.5.5 | Will applicable aspects of the SCM plan, environment, tools, training and procedures be conveyed to any sub-tier suppliers of components of the system and subcontractors to ensure their compliance to the approved plans, standards and procedures? | • A-8, #1-6 |
| **1.6** | **Review the SQA plan and consider the following questions:** | |
| 1.6.1 | If the SQA plan is followed, will the DO-178B objectives defined in Table A-9 be satisfied? | • A-9, #1-3 |
| 1.6.2 | Are the SQA processes as defined in Section 8.0 of DO-178B described in sufficient detail (see DO-178B Section 11.5) to satisfy Section 4.2? | • A-1, #1 |
| 1.6.3 | Does the SQA plan adequately address the proposed contents described in DO-178B, Section 11.5? If not, are the contents included in another plan? | • A-1, #1 |
| 1.6.4 | Is SQA independent from the development organization to a sufficient degree to ensure that SQA has the autonomy and authority to ensure SQA audit findings, actions, and deficiencies will be corrected? | • A-1, #2<br>• A-9, #2 |
| 1.6.5 | Are there any deviations proposed for this project from the SQA plans and procedures? If so, are those deviations identified and justified? | • A-1, #2<br>• A-9, #2 |
| 1.6.6 | Are the transition criteria, interrelationships, and sequences among process properly and adequately defined, and are they capable of being audited to ensure process compliance? | • A-1, #2<br>• A-9, #2 |
| 1.6.7 | Are there defined procedures for how SQA audit findings, actions, and observed deficiencies will be corrected for the project? | • A-1, #2<br>• A-9, #2 |
| 1.6.8 | Is the SQA findings process a separate process or does it use different tools for issues resolution than the development organization (i.e., how are the SQA and development team problems reported)? | • A-1, #2<br>• A-9, #2 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.6.9 | Is the criteria for SQA involvement (sampling, attending reviews, evaluation, conducting process compliance audits, witnessing tests, "conforming" environments, etc.) defined for the project? (Note: Criteria for involvement may be dependent on the software level or novelty of the product being developed.) | • A-1, #2<br>• A-9, #1-2 |
| 1.6.10 | Has an accountable person or organization been identified for each documented SQA process and activity? | • A-1, #1<br>• A-9, #1-3 |
| 1.6.11 | Will applicable aspects of the SQA plan, environment, tools, training, and procedures be conveyed to any sub-tier suppliers of components of the system and subcontractors to ensure their compliance to the approved plans, standards, and procedures? | • A-1, #1-3, 7<br>• A-9, #1-2 |
| **1.7** | **Review the SVP and consider the following questions:** | |
| 1.7.1 | If SVP is followed, will objectives of A-3, A-4, A-5, A-6, and A-7 be met? | • A-3 to A-7 (all objs) |
| 1.7.2 | Does the SVP adequately address the proposed contents described in DO-178B, Section 11.3? If not, are the contents included in another plan? | • A-1, #1-3 |
| 1.7.3 | Will applicable aspects of the SVP plan, environment, tools, training and procedures be conveyed to any sub-tier suppliers of components of the system and subcontractors to ensure their compliance to the approved plans, standards, and procedures? | • A-1, #1-3, 7<br>• A-3 to A-7 (all objs) |
| 1.7.4 | Does the SVP describe how independence will be achieved, when required? | • A-3 to A-7 (all objs) |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.7.5 | Does the SVP describe the verification method used for each software verification activity? Specifically:<br>• Are methods, checklists, tools and procedures described for conducting reviews of software requirements, design, coding, and integration?<br>• Are methods, checklists, tools and procedures described for conducting analyses of traceability, change impact, timing, memory usage, stack usage, common shared resource (memory, I/O ports, buffers and devices, floating point processor, cache, etc.) usage, requirements-based test coverage, structural coverage, normal range coverage, robustness test coverage, data coupling, control coupling, etc.?<br>• Are methods, checklists, tools and procedures described for conducting reviews of test plans, test procedures, test cases, and test results?<br>• Are methods, checklists, tools and procedures described for conducting testing of software high-level requirements, software derived requirements, software low-level requirements, software components, software integration, hardware-software integration, normal range, and robustness?<br>• Will most of the formal software verification testing be conducted on a "host" computer environment or on the target environment? Note: If conducted on a host, justification should be provided for why the testing is valid for the target environment.<br>• Will most of the formal software verification testing be conducted on the executable object code embedded in the target environment, or on another form of the software (e.g., assembly language) on a "host" computer environment? Note: If conducted on software other than the final airborne software load, justification should be provided for why the testing is valid for the airborne software in the target environment.<br>• If software verification test credit will be claimed for testing conducted on system benches, laboratory, integrated system facilities, do the plans and procedures describe how those activities will be conducted and software test results and coverage analyses documented?<br>• Is there a well-defined process and procedure for ensuring that deficiencies detected during the testing process will be conveyed to and corrected by the software development process and team? | • A-1, #1-3 |
| 1.7.6 | Does the SVP describe the verification environment, including the test equipment? Consider the following questions:<br>• Are there any automated tools? If so, do any of the tools need to be qualified?<br>• Is there any overlap between various kinds of testing (e.g., overlap of system and requirements-based testing)?<br>• Is the division of the testing task between suppliers and sub-contract suppliers adequately addressed and controlled? | • A-1, #1-3 |
| 1.7.7 | Are verification plans, including test plans, and procedures conveyed to suppliers and sub-contractors to ensure their activities and results will comply with the approved plans and procedures? | • A-1, #1-3 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.7.8 | Do the plans and procedures describe how configuration control of the software test environment will be maintained? If a system test environment will be used for formal software testing "credit," do the plans and procedures describe how the system test bench will be configuration controlled and conformed? | • A-1, #1-3 |
| 1.7.9 | Does the SVP describe methods for test case selection? Does the SVP specify how each requirement will be tested (e.g., module test, software integration, hardware-software integration, system, etc.)? | • A-1, #1-3 |
| 1.7.10 | Does the SVP or procedures specify who is allowed to perform verification tasks? | • A-1, #1-3 |
| 1.7.11 | For Levels A, B, and C software, do the plans address all aspects of structural coverage analysis? For example, are the following addressed:<br>• tools and tool qualification, if tools are used for structural coverage analysis and results recording<br>• the relationship between requirements-based testing and measuring structural coverage<br>• a process for determining when additional requirements-based tests should be added if coverage is not achieved as expected<br>• a procedure for regression analysis and testing, if necessary<br>• the transition criteria to start and end structural coverage analysis<br>• regression analysis and testing with respect to the unique requirements for structural coverage<br>• processes and procedures for conducting analyses of data coupling (data interfaces and dependencies between system components) and control coupling (execution interfaces and dependencies between system components)<br>**NOTE:** See *A Practical Tutorial on Modified Condition/Decision Coverage* [3] for additional information on MC/DC. | • A-1, #1-3<br>• A-7, #5-8 |
| 1.7.12 | If it is level A software and structural coverage analysis is performed at the source code level, does the SVP address how source to object code traceability will be performed and documented?<br>**NOTE:** See CAST-12 [4] for additional information on source to object code traceability. | • A-1, #1-3<br>• A-7, #5 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.7.13 | If verification tools are used, consider the following questions to determine whether the tool(s) needs to be qualified:<br>• Does the tool eliminate, reduce or automate a process or activity related to compliance with DO-178B?<br>• Can the verification tool allow an existing error to remain undetected? If so, what classes of errors can the tool fail to detect? Is there another verification activity performed to detect these classes and instances of errors?<br>• Is the output of the verification tool(s) verified manually or by another tool?<br>**NOTE:** See section 12.2 of DO-178B and chapter 9 of Order 8110.49 [2] for more information on tools. | • A-1, #1-4 |
| 1.7.14 | If verification tools are reused, does the SVP (or other document) address possible reuse of verification tools? For example, is credit being claimed from previous tool qualifications or will the tool qualification data be used in a future program? | • A-1, #1-4 |
| **1.8** | **Develop an understanding of the system from applicant's plans, safety assessment, standards, and briefings.** | |
| 1.8.1 | Does the system safety assessment, system architecture, and safety features support the software level for every software component, as proposed in the plans? | • A-1, #1 |
| 1.8.2 | Are SAE ARP-4754 and/or ARP-4761 the proposed means of compliance for conducting system development process and system safety assessment process, respectively, or does the applicant/developer have other proposed means of compliance? | • A-1, #1 |
| 1.8.3 | Are safety features to be implemented in software identified to the software processes as safety-related requirements? | • A-1, #1, 2 |
| 1.8.4 | Are safety features to be provided by the system or hardware robust enough to support the assigned software levels? | • A-1, #1 |
| 1.8.5 | Are derived software requirements provided back to the system safety assessment process for analysis of their impact on the safety assessment? | • A-1, #1, 2 |
| **1.9** | **Review the software development standards and consider the following questions:** | |
| 1.9.1 | Are the software development standards (requirements, design, and coding) identified and well defined? | • A-1, #5 |
| 1.9.2 | Are the software development standards consistent with the plans and do they support implementation of the plans? | • A-1, #5 |
| 1.9.3 | Will the software development standards support the proposed software level(s) and software's compliance with DO-178B objectives and other applicable policy and guidance? | • A-1, #5 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.9.4 | Have standards been verified for each defined software life cycle process? Are the standards adequate to support the software level? | - A-1, #5 |
| 1.9.5 | Have standards been verified to ensure compliance to Section 11 of DO-178B (i.e., sections 11.6, 11.7, and 11.8)? | - A-1, #5 |
| 1.9.6 | Have standards been verified to ensure that any constructs are not permitted which would invalidate the assumptions about the safety levels (e.g., unconstrained recursion, non-determinism)? | - A-1, #5 |
| 1.9.7 | Have the software development standards been verified to ensure that there are limitations, prohibitions, and constraints to not permit the use of design and code features that are not deterministic and not verifiable? | - A-1, #5 |
| 1.9.8 | Do design standards address exception handling, interrupts, re-entry, recursion, and scheduling constraints and methods? (See DO-178B, Section 11.7) | - A-1, #5 |
| **1.10** | **Review the plans to determine if real-time aspects of the software implementation have been addressed. Consider if the following questions have been addressed in the plans:** | |
| 1.10.1 | Is the processor type identified in the plans? Does the applicant have experience with the processor? Is the processor usage justified? Additionally, consider the following questions:<br>- What features of the processor are unique and/or may impose specific scheduling, priorities, memory management, protection, and so forth requirements on the software?<br>- What safety features and capabilities does the processor have that will be used by the system to support system requirements? Will these features have an impact on the design of the software?<br>- Does the developer have plans and procedures for how they will use the features of the processor to develop the software functions' architecture, scheduling, priorities, partitioning, built-in test, memory management, etc.? If so, how will those features and capabilities will be verified? | - A-1, #3 |
| 1.10.2 | Will the processor cache and/or pipelining be used? If so, how will partitioning, isolation, separation, and other protection methods and worst-case execution timing for the system be addressed? | - A-1, #1, 3 |
| 1.10.3 | Does the system use a board support package or device drivers? Are any of them new, modified, or unique for this system? How will the applicant or developer show that they satisfy the DO-178B objectives or guidance of DO-254? Is any service history credit claimed for their use? | - A-1, #3 |
| 1.10.4 | Is a memory management unit (MMU) used? If so, do the plans address how its accuracy and correct function will be verified? | - A-1, #1, 3 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.10.5 | Does the system include any libraries, functions, macros, etc. provided by the compiler, real-time system, operating system, microprocessor, or development environment vendors? Are plans and procedures established for how these components will be shown to satisfy the DO-178B objectives and other applicable software policy? Does the applicant have a user's guides for these components that describe their intended functions and any constraints or limitations for their use? | • A-1, #1, 3 |
| 1.10.6 | Are any application programming interfaces (APIs) used? If so, are the following addressed in the plans:<br>• Are their functions described?<br>• Can they be fully verified?<br>• Can they have any effects on the other applications and functions of the system?<br>• Are there requirements that specify the interfaces in detail (e.g., user's guide or interface specification)?<br>• Does the system or system's software applications have the ability to detect any errors of the API to be used? | • A-1, #3 |
| 1.10.7 | Is an integrated development environment (IDE) used? Will any of the tools be qualified? Are dependencies and interactions between the tools of the environment document and well understood? Are the tools in the IDE compatible with one another and with other tools used in the development and verification processes? Is this well documented? | • A-1, #3 |
| 1.10.8 | Do the plans describe how the development environment will be preserved for any future changes to software? Does the applicant or developer have plans and procedures for managing changes to the tools and analyzing their impact on already approved as well as future systems and projects? | • A-1, #3, 6 |
| 1.10.9 | Are cyclic redundancy checks (CRC) used? If so, how is their accuracy and adequacy ensured? Are CRC's or checksums used for loading software, for verifying databus message content correctness, for verifying files stored in system memory, for conducting built-in testing, etc.? Are other checksums and parity checks used? Are plans and procedures established for ensuring that CRC's, checksums, parity checks, and the like provide the appropriate level of integrity and assurance for each use? | • A-1, #1 |

| Item # | SOI #1 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 1.10.10 | What language is being used?<br>• Does the applicant have experience with the language?<br>• Does the language have any dynamic features or capabilities that may be difficult to verify? Has the applicant or developer imposed any prohibitions or restrictions on the use of these features in the coding standards and/or review checklists? Does the applicant/developer have plans and procedures established for how these dynamic features will be verified?<br>• What features of the language support or hinder real-time implementation and verification?<br>• Are any language problems documented and addressed?<br>• Has a safe sub-set of the language been selected? How will it be verified that that safe sub-set was adhered to (e.g., code reviews, checklists, testing, etc.)?<br>• What are common errors of the language? Do coding standards protect against these errors? | • A-1, #3, 5 |
| 1.10.11 | If the system or its software will use multitasking, what kind of scheduling strategy is used? How will priorities, protection, and interrupts be addressed? | • A-1, #1, 6 |
| 1.10.12 | If partitioning or other protection capabilities are used, how will they be implemented and verified? | • A-1, #1, 6 |
| 1.10.13 | If a Real-Time Operating System (RTOS) is used:<br>• How will the applicant demonstrate that it satisfies the DO-178B objectives and other applicable software policy? Will service history credit be claimed for its use in this system? If so, is the service history relevant for the system being developed?<br>• How will RTOS requirements trace to the software applications and/or system requirements?<br>• Will there be any unused features of the RTOS? If so, how will it be addressed (either not loaded in the airborne application or shown to be deactivated)?<br>• Will the RTOS be used to implement multiple software levels of software applications? If so, has the applicant or developer addressed partitioning and protection capabilities in their plans and procedures for time partitioning (shared use of the CPU, registers, cache, coprocessors, I/O devices, and other "shared" system resources) and space (memory, such as, shared use of RAM, stack space, I/O buffers, global data and other shared memory devices) been addressed? How will these features and capabilities be verified? | • A-1, #1,3,6 |
| 1.10.14 | Are there any libraries used in the software? If so, how will the applicant show that they satisfy the DO-178B objectives and other applicable software policy and agreements? | • A-1, #1,3,6 |

| | | |
|---|---|---|
| **1.11** | **If object-oriented technology is used.  Consider if the following questions have been addressed:** | |
| 1.11.1 | Does the additional considerations section (or some other section) of the PSAC address that the project is using object-oriented (OO) technology and identify OO-related issues for the specific project?  Consider the following questions:<br>• Are there specific standards and guidance applicable for the project using OO?<br>• Does the applicant or developer have experience using OO for airborne system developments?<br>• Do the DERs for the project have experience assessing OO projects and data to compliance with DO-178B? | • A-1, #4 |
| 1.11.2 | Do the plans address how the following common OO issues will be addressed?<br>• Encapsulation<br>• Overloading<br>• Inheritance<br>• Dynamic binding/dispatch<br>• Polymorphism<br>• Templates<br>• Inlining<br>• Dead pr deactivated code<br>• Traceability<br>• Structural coverage<br><br>Will verification of the OO development have specific procedures, testing, and analyses to address the above issues?  Does the applicant or developer have experience in addressing these issues for past projects/systems?<br><br>**NOTE 1:** CAST-8 [5] addresses common C++ issues, which may also apply to other OO languages.<br>**NOTE 2:** The FAA is developing a handbook on OO issues – it is slated for release in June 2004.  Volume 4 of the handbook will contain questions that should be considered in addition to this Job Aid. | • A-1, #1-7 |

## *NOTES:*

## 3.2 ACTIVITIES FOR STAGE OF INVOLVEMENT #2 – DEVELOPMENT REVIEW

***Purpose***

- Assess effective implementation of applicant’s plans and standards through examination of software life cycle data.
- Assess and agree to any changes in the plans.
- Assure that software life cycle data meets DO-178B objectives from tables A-2, A-3, A-4, A-5, A-8, A-9, and A-10.

***When Review Occurs***

When the software design is sufficiently mature to support ongoing software change without degrading safety or architecture, or when deemed necessary. Some things that should be in place prior to this review are:

- High-level requirements are documented, reviewed, and traceable to system requirements.
- Low-level requirements are documented, reviewed, and traceable to high-level requirements.
- Source code implements and is traceable to the low-level requirements and has been reviewed.

***Data Reviewed Prior to Review***

Report from Stage of Involvement #1 (SOI #1); open items from SOI #1; and all plans (PSAC, SVP, SDP, SCMP, SQAP (as a refresher after SOI #1 or to review changes to plans since SOI #1)).

***Data Reviewed at Review***

Standards for Software Requirements, Design, and Code; Software Requirements Data; Design Description; Source Code; Software Verification Results (as applied to Tables A-2 to A-6); Problem Reports; Software Configuration Management Records; Software Quality Assurance Records; Trace Matrix/Tool; and Designees’ findings/observations from pre-review activities.

***Agenda***

See Supplement 3, Section 1.

***Number of Days Required***

2-3 days

***Evaluation Activities and Questions***

During Stage of Involvement #2 (SOI #2), a trace matrix or similar traceability tool should be used to perform one or two top-down traces (from system requirements to high-level software requirements to design description to code to test cases) and one or two bottom-up traces (from code to design description to software requirements to systems requirements) in each of the major sub-systems. The traces should be performed in different areas (e.g., display, interface, core logic). During the traces the following activities/questions listed in SOI #2 table below should be addressed.

***Instructions***

- There are sixteen major evaluation activities for Stage of Involvement #2: Development Review.
- Review the questions for each activity in relationship to its corresponding DO-178 objective and software criticality.

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| **2.1** | **Analyze high-level requirements and associated derived high-level requirement(s) traceability to the selected system level requirement.** | |
| 2.1.1 | Are all software high-level and derived high-level requirement(s) associated with each selected system level requirement (i.e., identify threads between system and high-level requirements) clearly identified? (Use of a development trace matrix/tool will be helpful.) | • A-3, #1, 6 |
| 2.1.2 | If independence is required for the software level, is the person doing the verification different than the one responsible for developing the requirement(s)? | • A-3, #1<br>• A-4, #1 |
| 2.1.3 | Is each requirement uniquely identified (i.e., each requirement number is truly only one requirement)? | • A-3, #4 |
| 2.1.4 | Are the requirements unambiguous? Does the requirement have the same meaning to all participants (acquirer, systems engineer, software developers, and users)? | • A-2, #2 |
| 2.1.5 | Are requirements consistent (e.g., terminology attributes, data definitions)? | • A-3, #2 |
| 2.1.6 | Are high-level software requirements (that trace back to that system level requirement) accurate? That is, if all these high-level requirements are met, would the associated system level requirement be satisfied? If not, determine if there are additional hardware requirements that if implemented, the combination of software and hardware requirements would satisfy the corresponding high-level requirement. | • A-3, #2 |
| 2.1.7 | Are the requirements complete? Are there any “To Be Determined” items in the requirements data? | • A-3, #1 |
| 2.1.8 | Is each requirement verifiable through testing, inspection, or analysis? (Note: Timing, sizing, and partitioning requirements are generally supported through analysis.) | • A-3, #4 |
| 2.1.9 | Does each requirement conform to standards as defined in the developer’s software requirements standards? | • A-3, #5 |
| 2.1.10 | Have requirements been reviewed to determine that algorithms are accurate (see DO-178B table A-3, objective 7)? (Note: This may require a math background or in-depth knowledge of the system.) | • A-3, #7 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.1.11 | If using table A-3 and the requirements verification results, have the objectives for each selected requirement(s) been previously verified? (See DO-178B Section 11.14 for details on verification results.) | • A-3, #4 |
| 2.1.12 | Are there inconsistencies between the data reviewed and the software development plans? | • A-3, #2 |
| 2.1.13 | Do any conversations with developers indicate that the plans were not followed? (Determine through interview/discussions with developers.) | • A-9, #1 |
| 2.1.14 | If real-time operating system (RTOS), board support package (BSP), application programming interface (API), or device drivers are used, are the requirements and interfaces identified and traced? | • A-2, #1-5<br>• A-3, #6<br>• A-4, #6 |
| 2.1.15 | Is the use of "derived requirements" used accurately? (see last paragraph of DO-178B Section 5.0 for definition of derived requirements). | • A-2, #2, 5 |
| 2.1.16 | Do verification results exist to demonstrate verification of all applicable table A-3 objectives? Were the verification activities thorough and well documented? | • A-3, #1-7 |
| **2.2** | **Review the software design and design data and determine compliance to DO-178B Table A-4.** | |
| 2.2.1 | Is each low-level requirement(s) and associated derived low-level requirement(s) traceable to each high-level requirement examined during the requirements review process? (Reference DO-178B Table A-4.) | • A-4, #1, 6 |
| 2.2.2 | Is there traceability between the high-level and low-level requirements? This is usually evaluated by using the trace tree/matrix. | • A-4, #6 |
| 2.2.3 | Is the selected high-level requirement(s) reflected in the design? (i.e., does the design implement the high-level requirement to which it traces?) | • A-4, #6 |
| 2.2.4 | Are the low-level requirements uniquely identified (i.e., each requirement number is truly only one requirement)? | • A-4, #4 |
| 2.2.5 | Are the low-level requirements unambiguously stated (one meaning by developer, user, and acquirer) with clear concise and consistent terminology? | • A-4, #2 |
| 2.2.6 | Are performance requirements (timing, sizing, throughput) identified? | • A-4, #10 |
| 2.2.7 | Are fail-safe, fail-operational requirements specified (when applicable)? | • A-4, #12 |
| 2.2.8 | Are all low-level software requirements (that trace back to each high-level requirement) accurate? | • A-4, #2 |
| 2.2.9 | If the low-level requirements (traceable to the high-level requirement) are implemented correctly, would the high-level requirement be met? | • A-4, #1, 6 |
| 2.2.10 | Is each low-level requirement verifiable through testing, inspection, or analysis? | • A-4, #4 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.2.11 | Has appropriate analysis of verification results of the software low-level requirements and the software architecture taken place (DO-178B paragraph 6.3.15, 6.3.3)? Using table A-4 and the design verification results, assure that the objectives for each identified (i.e., in the trace tree) low-level requirement have been previously verified. See DO-178B Section 11.14 for details on verification results. | • A-4, #1-12 |
| 2.2.12 | Does each low-level requirement conform to developer's design standards? | • A-4, #12 |
| 2.2.13 | Is the algorithm associated with the selected low-level requirement accurate? (This may require a math background and/or system knowledge.) | • A-4, #7 |
| 2.2.14 | Are units specified consistently? | • A-4, #5 |
| 2.2.15 | Are there any inconsistencies between the data reviewed and the software development plans? | • A-9, #1 |
| 2.2.16 | Do any conversations with developers indicate that the plans were not followed? (Determine through interview/discussion with developers.) | • A-9, #1 |
| 2.2.17 | Do verification records exist to demonstrate verification of all applicable table A-4 objectives? Were the verification activities thorough and well documented? | • A-4, #1-13 |
| **2.3** | **Review the software architecture.** | |
| 2.3.1 | Is the partitioning schema sufficient to support the high-level requirements and the software level established by the system safety assessment? | • A-4, #8-12 |
| 2.3.2 | Has partition integrity (i.e., protection) been achieved (in terms of time, space, and throughput)? | • A-4, #13 |
| 2.3.3 | Does the interrupt/control structure support the known system priorities and high-level requirements? | • A-3, #3 |
| 2.3.4 | Does the architecture support the timing and sizing requirements? | • A-4, #10,11 |
| 2.3.5 | Are the synchronous vs. asynchronous aspects of the design supported by the architecture? | • A-4, #10 |
| 2.3.6 | How is exception handling addressed? | • A-4, #9 |
| 2.3.7 | Are data flows consistent? | • A-4, #9 |
| 2.3.8 | Are interfaces consistent? | • A-4, #10 |
| 2.3.9 | Does the communication mechanism support the high-level requirements? | • A-4, #8 |
| 2.3.10 | Is the architecture sufficient to provide service to time-critical tasks? | • A-4, #10 |
| 2.3.11 | Does the architecture conform with design standards? | • A-4, #12 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.3.12 | Has software architecture been reviewed and analyzed to address compatibility with target computer? | • A-4, #10 |
| **2.4** | **Review the software code/integration data to determine if objectives of DO-178B Table A-5 are met.** | |
| 2.4.1 | Does traceability exist between the code and the software low-level requirement? | • A-5, #1, 5 |
| 2.4.2 | Does the calling sequence correspond with the software architecture? | • A-5, #2 |
| 2.4.3 | Does the source code have to be altered to test it? | • A-5, #3 |
| 2.4.4 | Are the data definitions correct? Consider the following criteria:<br>• Data typing is correct and consistent.<br>• Units are consistent between modules (e.g., radians, degrees).<br>• All variables used are also defined.<br>• Data are properly initialized.<br>• Global data integrity is assured.<br>• Variables are not used for more than one purpose. | • A-5, #4, 6 |
| 2.4.5 | Does the source code conform to standards? Consider the following areas typically found in standards:<br>• Is indentation schema being followed?<br>• Are prologue headers per the standards?<br>• Are naming conventions following the standards?<br>• Are the size of the modules per the standards?<br>• Does the code do what the comments say it does?<br>• Is there only one entry and exit point?<br>• Are only the standard coding constructs as defined in the coding standards used?<br>• Are nesting considerations being addressed? | • A-5, #4 |
| 2.4.6 | Has computational correctness been achieved? Consider the following criteria:<br>• Sign conventions are consistent and correct.<br>• Precision is maintained in mixed mode arithmetic.<br>• Desired accuracy is maintained during rounding or truncation.<br>• Divide by zero is prohibited/ trapped. | • A-5, #6 |
| 2.4.7 | Are the logic constructs and data handling correct? Consider the following criteria:<br>• Loops are correctly implemented.<br>• Subscripts are used properly. | • A-5, #4 |
| 2.4.8 | Were issues identified and supported with issue papers? | • A-1, #4 |
| 2.4.9 | Do verification records exist to demonstrate verification of all applicable table A-5 objectives? Were the verification activities thorough and well documented? | • A-5, #1-7 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| **2.5** | **Determine if the requirements and design have been reviewed, considering the following real-time questions:** | |
| 2.5.1 | Have high-level requirements been reviewed and/or analyzed to identify any conflicts on the target computer (for Levels A and B)?<br>• Are system response times addressed?<br>• Is the Input/Output (I/O) adequate?<br>• Have reviews/analyses been documented? | • A-3, #3 |
| 2.5.2 | Have low-level requirements been reviewed and/or analyzed to identify any conflicts on target computer (for Levels A and B)?<br>• Are system response times addressed?<br>• Are resource issues addressed?<br>• Is the I/O adequate?<br>• Have reviews/analyses been documented? | • A-4, #3 |
| 2.5.3 | Has software architecture been reviewed and/or analyzed to identify any conflicts on the target (for Levels A and B)?<br>• Have initialization issues been addressed?<br>• Have synchronization issues been addressed?<br>• Are interrupts properly supported? | • A-4, #10 |
| 2.5.4 | Are performance requirements supported by the hardware? Such as:<br>• Timing?<br>• Sizing?<br>• Throughput? | • A-4, #10 |
| **2.6** | **Determine if the real-time aspects of the system development have been addressed. Consider the following questions:** | |
| 2.6.1 | Are run-time libraries used? If so:<br>• Are the libraries specified?<br>• Do the requirements, design, and code exist for the used library functions?<br>• Will structural coverage be applied on the libraries or just on features used by the application program?<br>• How is code not used by the application dealt with?<br>• Is there dead code in the libraries?<br>• How are problems found in the library routine dealt with by the library developer and/or the applicant?<br>• Have the libraries been verified? | • A-2, #1-6 |
| 2.6.2 | Are requirements and design for time-critical tasks specified in quantifiable terms? | • A-3, #4<br>• A-4, #4, 11 |
| 2.6.3 | Do software requirements and design address timing constraints, strategy for dealing with timing limits, required timing margins, method of measuring timing margins? | • A-2, #3-5 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.6.4 | If used, is error prevention, fault tolerance, or error detection specified in the requirements, design, and code? | • A-2, #1-6 |
| 2.6.5 | If interrupt service routines (ISRs) are used, are they documented in the requirements/design? Do they work properly? Specifically, does the ISR:<br>• Block any continuing execution?<br>• Call reentrant functions?<br>• Pass stress testing?<br>• Allow calls to functions before completing? | • A-2, #3-5 |
| 2.6.6 | How have the common concurrency problems been addressed, such as:<br>• Deadlock,<br>• Livelock,<br>• Race conditions,<br>• Re-entrancy,<br>• Priority inversion,<br>• Mutual exclusion violation, and<br>• Non-deterministic execution order. | • A-2, #1-6 |
| 2.6.7 | Is partitioning/protection used? If so, is it documented in requirements and design? | • A-2, #1-6<br>• A-4, #13 |
| 2.6.8 | How is synchronization and communication addressed in the system (e.g., synchronous or asynchronous)? Are the synchronization and communication mechanisms documented in the requirements and design data? | • A-2, #1-6 |
| 2.6.9 | If buffers are shared, has the reader-writer (producer-consumer) problem been addressed? | • A-2, #1-6 |
| 2.6.10 | Have the following common communication problems been addressed by the applicant:<br>• Lost data,<br>• Stale data,<br>• System hanging,<br>• Bounded buffer, and<br>• Corrupted data? | • A-2, #1-6 |
| 2.6.11 | Are critical sections protected? How are they protected? Is the protection adequate and accurately implemented? | • A-4, #13 |
| 2.6.12 | What kind of scheduling algorithm has been selected for the real-time system? Is the algorithm documented in the requirements and design? Is the scheduling algorithm deterministic and verifiable? | • A-2, #1-6 |
| 2.6.13 | If the scheduler uses priorities:<br>• How are priorities determined?<br>• What happens when two tasks have the same priority?<br>• How has priority inversion been addressed?<br>• How are interrupts handled? | • A-2, #1-6 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.6.14 | If concurrent tasks are run, how are they handled? (I.e., Is multitasking used?) What algorithms are used to implement concurrency? Are threads used? If threads are used, how do they affect timing? | • A-2, #1-6 |
| 2.6.15 | Is there a mechanism to detect when real-time tasks do not meet their deadlines? If detected, what is the response and is it consistent with the safety requirements? | • A-2, #1-6 |
| 2.6.16 | Has a tool been used to document the schedule (e.g., rate monotonic analysis tool)? Has the output of the tool been verified? If not, has it been qualified? | • A-2, #1-6 |
| **2.7** | **Review the configuration management data to determine compliance to DO-178B Table A-8.** | |
| 2.7.1 | Review the Configuration Identification and consider the following questions: | |
| 2.7.1.1 | Are all software life cycle data uniquely identified? | • A-8, #1 |
| 2.7.1.2 | Does configuration identification provide for the identification of the product, components, sub-components – both individually and collectively? | • A-8, #1 |
| 2.7.1.3 | Does configuration identification of data occur prior to use of that data by another life cycle process? | • A-8, #1 |
| 2.7.2 | Review Baselining Activity and consider the following questions: | |
| 2.7.2.1 | Does the product baseline include the following data:<br>• PSAC, Configuration Index, and the Accomplishment Summary?<br>• Software requirements data?<br>• Each source code component?<br>• Previously developed software, if used in the software product?<br>• Instructions for building the executable code to include compiling and linking?<br>• Software life cycle (development) environment?<br>• Executable Code? | • A-8, #2 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.7.2.2 | Does the development include the following in the product baseline:<br>• Software Development, Verification, Configuration Management, and Quality Assurance Plans?<br>• Requirements, Design, and Code Standards?<br>• Design Description?<br>• Software Verification Cases and Procedures? | • A-8, #2 |
| 2.7.3 | Are tools used in the development environment under configuration control? | • A-8, #6 |
| 2.7.4 | Are the compiler options controlled and settings included in software life cycle environment configuration index (SLECI)? | • A-8, #6 |
| **2.8** | **Review the Problem Reports and changes to software life cycle data for impact on software code.** | |
| 2.8.1 | Has the applicant documented problems as described in their plans? (e.g., problem reports) | • A-8, #3 |
| 2.8.2 | Is there a problem reporting process in place? Is there a change control process? Is the applicant following the processes? | • A-8, #3 |
| 2.8.3 | Does the problem report adequately describe the deficiency or anomalous behavior and the proposed change(s)? | • A-8, #3 |
| 2.8.4 | Are all affected software module(s) identified? | • A-8, #3 |
| 2.8.5 | Was the configuration updated to reflect the new version(s)? | • A-8, #3 |
| 2.8.6 | Does the change made to the software support the problem report description? | • A-8, #3 |
| 2.8.7 | Was the correct form used? | • A-8, #3 |
| 2.8.8 | Was change authorization properly confirmed? | • A-8, #3 |
| 2.8.9 | Was the change documented in the header? | • A-8, #3 |
| 2.8.10 | If the change affected the software design was the design data updated or a change applied to the baseline? | • A-8, #3 |
| 2.8.11 | If the change affected the requirements, were the requirements updated or a change applied to the baseline? | • A-8, #3 |
| **2.9** | **Review of Archival, Retrieval, and Release Procedures.** | |
| 2.9.1 | Was the product protected from unauthorized changes? | • A-8, #4 |
| 2.9.2 | Does the storage medium minimize risk of deterioration and regeneration of errors? | • A-8, #4 |
| 2.9.3 | Are copies stored in physically separate archives for disaster recovery? | • A-8, #4 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.9.4 | Are there provisions to make and verify error free copies? (Includes executable document name/version/date, paragraph numbers, requirements identification, and results.) | • A-8, #4 |
| **2.10** | **Review the Software Quality Assurance Data to assure that the objectives of DO-178B Table A-9 are met.** | |
| 2.10.1 | Was the quality assurance plan followed? | • A-9, #1 |
| 2.10.2 | Is there objective evidence that the transition criteria were satisfied prior to transitioning to the next phase(s) of development? | • A-9, #2 |
| 2.10.3 | Were any of the reviewed processes or products deficient? If so, examine the software quality assurance records and speak with the quality assurance manager in order to:<br>• Determine why it was not found earlier.<br>• If the process in place was not effective enough to detect the problem, ask what corrective action will be taken.<br>• Assess possible causes of quality assurance function deficiencies (e.g., overworked staff, staff lacks knowledge). | • A-9, #1 |
| 2.10.4 | What method was used to ensure that the version of the software verification result matches the version of the software executable code? (i.e., was a conformity review conducted?) | • A-9, #3 |
| **2.11** | **Optional: Review sampling of the applicant's test cases and procedures (even if they are in preliminary format). Review for adequacy of the test cases. The actual activities and questions for review of test cases and procedures is in SOI #3; however, it is a good practice to give some initial feedback to the applicant on test case development.** | • A-6, #1-5 |
| **2.12** | **Determine if the memory management has been adequately addressed. Consider the following questions:** | |
| 2.12.1 | How is memory managed (e.g., fixed partitioning, dynamic partitioning, simple paging, simple segmentation, virtual paging/segmentation)? Is memory managed in a deterministic manner? | • A-2, #1-6 |
| 2.12.2 | If shared memory is used, how has memory protection been defined and implemented? Has it been documented in the requirements and design? | • A-2, #1-6 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.12.3 | Do software requirements and design address memory size constraints, strategy for dealing with memory limits, required memory margins, and method of measuring memory margins? | • A-2, #1-6 |
| 2.12.4 | If a memory management unit is used, is it addressed in the requirements and design? | • A-2, #1-6 |
| 2.12.5 | Has the effect of the development environment on memory allocation been considered? | • A-2, #1-6 |
| 2.12.6 | Has memory leakage and fragmentation been evaluated? | • A-2, #1-6 |
| 2.12.7 | Is memory cache used? If so, has cache coherency been preserved? | • A-2, #1-6 |
| **2.13** | **Consider the following questions, if tools are used:** | |
| 2.13.1 | Is tool qualification needed? If so:<br>• Has a tool qualification plan been developed and reviewed?<br>• Has the tool qualification plan been followed?<br>• Has tool qualification data been developed and reviewed? | • A-2, #4<br>• Section 12.2 and applicable objectives |
| 2.13.2 | Are code generators used in the development? If so, is the output being verified? If the output is not being verified, is the tool being qualified? | • A-2, #6<br>• A-5, #1-7 |
| 2.13.3 | Are development tool constraints, notations, etc. addressed in the development standards? | • A-1, #5 |
| **2.14** | **If partitioning/protection is used, consider the following questions:** | |
| 2.14.1 | If partitioning/protection is needed, have both the time and space domains been considered? Has the partition/protection means been developed to the appropriate level? | • A-4, #13 |
| 2.14.2 | Is strict or safety, one-way or two-way protection used? | • A-4, #13 |
| 2.14.3 | What is the means used to separate memory? | • A-4, #13 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.14.4 | Have the following areas been considered for their effect on memory protection?<br>• Loss of input or output data<br>• Corruption of input or output data<br>• Corruption of internal data:<br>o direct or indirect memory writes<br>o table overrun<br>o incorrect linking<br>o calculations involving time<br>• Delayed data<br>• Program overlays<br>• Buffer sequence (double jeopordy)<br>• External device interaction (e.g. displays):<br>o loss of data (e.g. overwritten)<br>o delayed data<br>o incorrect data (unlikely across systems)<br>o protocol halts (e.g. ack nacks)<br>• Control Flow defects (space aspects):<br>o incorrect branching into a partition or protected area<br>o corruption of a jump table (double duty?)<br>o corruption of the processor sequence control<br>o corruption of return addresses<br>o unrecoverable hardware state corruption (e.g., mask and halt) | • A-4, #13 |
| 2.14.5 | Will different memory partitions be accessible at differed levels (i.e., is there shared data)? | • A-4, #13 |
| 2.14.6 | Will the same device drivers be used by the different level processes? Can a lower level corrupt the memory space associated with a device? | • A-4, #13 |
| 2.14.7 | Have the following real-time issues been evaluated for their affect on protection in the time domain:<br>• frame overrun<br>• interference with real time clock<br>• counter/timer corruption<br>• pipeline and caching<br>• loops (e.g., infinite loops)<br>• interrupts and interrupt inhibits | • A-4, #13 |
| 2.14.8 | Have the control flow defects been evaluated for their affect on time protection? E.g.,<br>• incorrect branching into a partition or protected area<br>• corruption of a jump table (double duty?)<br>• corruption of the processor sequence control<br>• corruption of return addresses<br>• unrecoverable hardware state corruption (e.g., mask and halt) | • A-4, #13 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.14.9 | Have the effects of software traps on time protection been considered? Such as,<br>o divide by zero<br>o un-implemented instruction<br>o specific software interrupt instructions<br>o unrecognized instruction | • A-4, #13 |
| 2.14.10 | How is time divided between functions (e.g., different criticality levels)? | • A-4, #13 |
| 2.14.11 | If cache is used and partitioning is required, how is memory partition addressed? | • A-4, #13 |
| 2.14.12 | Has a partitioning/protection analysis been performed to ensure the robustness of the partitioning/protection scheme? | • A-4, #13 |
| **2.15** | **If a RTOS is used consider the following questions:** | |
| 2.15.1 | How has the RTOS addressed data consistency? Is the approach adequate? Have the following data consistency concerns been addressed:<br>• Data corruption or loss<br>• Erroneous results<br>• Abnormal parameters | • A-2 to A-5 |
| 2.15.2 | Have common tasking concerns been addressed? Including:<br>• Inadvertent termination or deletion<br>• Kernel storage overflow<br>• Stack size exceeded | • A-2 to A-5 |
| 2.15.3 | Are the RTOS and CPU(s) interfaces well understood? Have the following RTOS/CPU interface questions been addressed:<br>• Has protected use of supervisor/user mode instructions been implemented?<br>• Is the memory model properly used and is memory properly allocated?<br>• Are caching algorithms deterministic, if used by the RTOS?<br>• Are there any microprocessor optimization features used by the RTOS?<br>• How does the CPU/RTOS handle hardware exceptions?<br>• Has the Board Support Package (BSP) been addressed?<br>• Is there code (new or previously developed) that may violate supervisor/user modes of the CPU? | • A-2 to A-5 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.15.4 | Have common scheduling concerns been addressed? Common concerns include:<br>• Corrupted task control blocks<br>• Excessive task blocking by priority inversion<br>• Deadlock<br>• Spawn tasks that starve CPU resources<br>• Corruption in task priority assignment<br>• Service calls with unbounded execution times<br>• Race conditions<br>• Specified maximum number of tasks<br>• Consistent use of RTOS schedules in the system | • A-2 to A-5 |
| 2.15.5 | Have memory and input/output concerns been addressed? Such as:<br>• Fragmentation of heap<br>• Incorrect pointer referencing<br>• Data overwrite<br>• Compromised cache coherency<br>• Memory locked<br>• Unauthorized access to devices<br>• Resource not monitored | • A-2 to A-5 |
| 2.15.6 | Have common queuing problems been addressed in the RTOS? Including:<br>• Task queue overflow<br>• Message queue overflow<br>• Kernel work queue overflow | • A-2 to A-5 |
| 2.15.7 | Have interrupt and exception concerns been addressed in the RTOS, such as:<br>• Interrupts during atomic operations<br>• No interrupt handler<br>• No exception handler<br>• Improper protection of supervisor task. | • A-2 to A-5 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 2.15.8 | If the RTOS is used to support partitioning/protection, have the following memory, I/O, and time partitioning/protection questions been addressed?<br>*Memory Partitioning:*<br>• How does the RTOS interact with the memory management unit (MMU)?<br>• How does the RTOS support partition protection?<br>• How is communications between partitions work?<br>• What caching model is used, if any?<br>• What action is taken if a memory partition is violated?<br>*I/O partitioning:*<br>• Is the I/O kernel centralized or handled within the partitions?<br>• Is there a task access permission mechanism used to access the I/O?<br>• What action is taken if an I/O partition is violated?<br>*Time partitioning:*<br>• How do the partition schedulers work and does there exist adequate margin to run the tasks in the partition?<br>• Is partition jitter handled effectively?<br>• How are task overruns or timing violations accommodated and how does the system react to a task overrun? | • A-4, #13 |
| 2.15.9 | Is the RTOS effect on worst-case execution time effectively calculated? | • A-2 to A-5 |
| 2.15.10 | Does the RTOS or system have a health monitor, and do the recovery mechanisms from problems meet the SSA? | • A-2 to A-5 |
| 2.15.11 | Is the RTOS User's Manual usable and up-to-date? Did the applicant use the User's Manual? | • A-2 to A-5 |
| 2.15.12 | How are unused parts of the RTOS addressed in the requirements? | • A-2 to A-5 |
| 2.15.13 | If RTOS was reverse engineered:<br>• Was the approach described in the plans followed?<br>• Does the reverse engineering approach meet DO-178B objectives?<br>• How were the high-level requirements created? Are they adequate? (Note: The low-level to high-level requirements transformation in reverse engineering is difficult and error-prone.)<br>• Does transition criteria and traceability exist? Are they accurate?<br>**Note:** See CAST-18 [6] for specific certification concerns regarding reverse engineering. In some cases, an issue paper may be required, if the plans do not address the DO-178B objectives compliance. | • A-2 to A-5 |

| Item # | SOI #2 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| **2.16** | **If object-oriented technology is used, have the following common issues been addressed in the development data?**<br>• Encapsulation<br>• Overloading<br>• Inheritance<br>• Dynamic binding/dispatch<br>• Polymorphism<br>• Templates<br>• Inlining<br>• Dead/deactivated code<br>• Traceability<br>**NOTE 1:** CAST-8 [5] addresses common C++ issues, which may also apply to other OO languages.<br>**NOTE 2:** FAA is developing a handbook on OO issues – it is scheduled for release in June 2004. Volume 4 of the handbook will contain questions that should be considered in addition to this Job Aid. | • A-2 to A-5 |

## 3.3 ACTIVITIES FOR STAGE OF INVOLVEMENT #3 – VERIFICATION REVIEW

***Purpose***

- Assess effective implementation of the applicant’s verification plans and procedures.
- Check the completion of all associated SCM and SQA tasks.
- Make determination on acceptable deviations from plans and standards found during the review or with the applicant’s requested deviations.
- Ensure that the selected software requirements have been verified.
- Ensure that the software life cycle data meets DO-178B objectives from Tables A-6, A-7, A-8, A-9, and A-10.
- Ensure that the verification activity satisfied the coverage requirements found in DO-178B, Table A-7.

***When Review Occurs***

When the following activities are in place (or when deemed necessary):

- Test procedures and results are documented and reviewed.
- Executable object code satisfies software requirements.
- A coverage analysis indicates the structural coverage requirements are met (for levels A, B, C).

Note: Testing is where many applicants have problems. It is best to perform this review early enough that major retest will not be required if issues are found, but late enough to see some trends in the test program.

***Data Reviewed Prior to Review***

Reports from SOI #1 and SOI #2, open items from SOI #1 and SOI #2, and all plans (PSAC, SVP, SDP, SCMP, SQAP (as a refresher after SOI #1 & SOI #2 or to review changes to plans since SOI #1 & SOI #2)).

***Data Reviewed at Review***

Software Requirements Data; Design Description; Source Code; Software Verification Cases and Procedures; Software Verification Results; Problem Reports; Software Configuration Management Records; Software Quality Assurance Records; Trace Matrix/Tool; and Designees' findings/observations from pre-review activities.

***Agenda***

See Supplement 3, Section 1.

***Number of Days Required***

2-3 days

***Evaluation Activities and Questions***

During SOI #3 a trace is performed on a sampling of system requirements allocated to software to see if they are adequately tested. Likewise, test cases are traced up to the system requirements to verify traceability. The table below describes some typical activities to perform and questions to ask during SOI #3.

***Instructions***

- There are fifteen major evaluation activities for Stage of Involvement #3: Verification Review.
- Review the questions for each activity in relationship to its corresponding DO-178B objective and software criticality.

| Item # | SOI #3 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| **3.1** | **Is there evidence that the SVP and other plans related to verification, integration, and testing  are being followed (e.g., progress against timeframes, staffing etc.)?** | • A-9, #1 |
| **3.2** | **Sample the applicant's test cases and consider:** | |
| 3.2.1 | Are test cases traceable to the requirements? | • A-6, #1-5 |
| 3.2.2 | Have normal ranges been tested? | • A-6, #1-5 |
| 3.2.3 | Has robustness testing been implemented? | • A-6, #1-5 |
| **3.3** | **Review test cases and procedures, considering the following questions:** | |
| 3.3.1 | Have test cases and procedures been reviewed for correctness? | • A-7, #1 |
| 3.3.2 | Do the test cases and procedures adhere to the relevant plans and standards? For example, have coding standards, especially those relevant to limitations of structural coverage tools, been followed? | • A-7, #1 |
| 3.3.3 | Are the test cases and procedures appropriately commented to allow future updates? | • A-7, #1 |
| 3.3.4 | Have the test cases and procedures been subjected to appropriate change and configuration control? | • A-7, #1 |
| 3.3.5 | Is the rationale for each test case clearly explained? | • A-7, #1 |
| 3.3.6 | Is the separation between test cases clear?  For example, are test start and stop identified?  This assists tracing the source of unexpected drops in coverage. | • A-7, #1 |
| 3.3.7 | Do the test cases and procedures specify required input data, expected output data, and input/output data (e.g., temporary stores)? | • A-7, #1 |
| 3.3.8 | Were the inputs for each test case derived from the requirements (as opposed to being derived from the source code)? | • A-7, #1 |
| 3.3.9 | Have the appropriate memory locations and variables been preset? | • A-7, #1 |
| 3.3.10 | Are the test cases and procedures sufficient to cover all the relevant requirements? That is, do the traceability matrices provide clear association between test cases and requirements? | • A-7, #1 |
| 3.3.11 | Are the test cases and procedures sufficient to meet MC/DC (for Level A)? | • A-7, #1 |
| 3.3.12 | Are sufficient tests to provide MC/DC identified for each logic construct (for Level A)? | • A-7, #1 |
| 3.3.13 | Are there sufficient robustness test cases and procedures? | • A-7, #1,3,4 |
| 3.3.14 | Are requirements where analysis is required in addition to (or in lieu of) requirements-based testing clearly documented (e.g., requirements for hardware polling)? | • A-7, #3-8 |

| Item # | SOI #3 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| **3.4** | **Review checklists for test cases, procedures, and results, considering the following questions:** | |
| 3.4.1 | Are the checklists sufficient to determine that the requirements-based test cases, procedures, and results meet verification objectives? | • A-7, #1, 2 |
| 3.4.2 | Do the checklists specify:<br>- who performed the review?<br>- what was reviewed (with revision data)?<br>- when it was reviewed?<br>- what was found?<br>- references to corrective actions, where necessary? | • A-7, #1, 2 |
| 3.4.3 | Will the procedure checklists reveal whether the results of the test cases that are counted for credit are observable? | • A-7, #1, 2 |
| 3.4.4 | Will the procedure checklists reveal test cases that violate project standards? | • A-7, #1, 2 |
| 3.4.5 | Will the procedures checklists reveal test cases that are not expected to achieve 100% structural coverage (e.g., hardware polling)? | • A-7, #1, 2 |
| 3.4.6 | Do the test case, procedures, and results checklists require evaluation of specified tolerances? | • A-7, #1, 2 |
| 3.4.7 | Have the test case, procedures, and results checklists been reviewed? | • A-7, #1, 2 |
| 3.4.8 | Do the test case, procedures, and results checklists ensure that test cases can be verified visually? | • A-7, #1, 2 |
| **3.5** | **Determine effectiveness of test program by: (1) assessing results of requirements-based tests, (2) assessing failure explanations and rework, and (3) assessing coverage achievement.** | |
| **3.5.1** | **Assess results of requirements-based testing, considering the following questions:** | |
| 3.5.1.1 | Are the test result files clearly linked to the test procedures and code? (i.e., does configuration control and traceability exist?) | • A-7, #2 |
| 3.5.1.2 | Is each test result clearly linked to a test case? | • A-7, #2 |
| 3.5.1.3 | Are failed test cases obvious from the test results? | • A-7, #2 |
| 3.5.1.4 | Do the test results indicate whether each procedure passed or failed and the final pass/fail results? | • A-7, #2 |
| 3.5.1.5 | Do the test results adhere to the relevant plans, standards, and procedures? | • A-7, #2 |
| 3.5.1.6 | Have the test results been subjected to appropriate configuration control? | • A-7, #2 |
| **3.5.2** | **Have all high-level and low-level requirements been tested?** | • A-7, #3, 4 |
| **3.5.3** | **Assess failure explanations and rework, considering the following questions:** | |

| Item # | SOI #3 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 3.5.3.1 | Is there an acceptable rationale for deviations from expected results, standards, or plans? | • A-7, #2 |
| 3.5.3.2 | Are explanations for the failed test cases intelligible? | • A-7, #2 |
| 3.5.3.3 | Do explanations for failed test cases contain accurate references to relevant problem reports? | • A-7, #2 |
| 3.5.3.4 | Are explanations for code or test rework suitable to address the failure? | • A-7, #2 |
| 3.5.3.5 | Have test cases been re-executed in compliance with plans for regression testing? | • A-7, #2 |
| 3.5.3.6 | Have the test results from regression testing been documented appropriately? | • A-7, #2 |
| **3.5.4** | **Assess structural coverage achievement, considering the following questions:** | |
| 3.5.4.1 | Is 100% structural coverage (as appropriate to the software level) achieved through requirements-based testing? | • A-7, #5-7 |
| 3.5.4.2 | If 100% structural coverage (as appropriate to the software level) is not achieved through requirements-based testing, is there an explanation detailing which parts of the code were not executed and why? | • A-7, #5-7 |
| 3.5.4.3 | Are explanations for drops in coverage sufficiently detailed and acceptable? | • A-7, #5-7 |
| 3.5.4.4 | Are there problem reports associated with dead code? | • A-7, #5-7 |
| 3.5.4.5 | Has dead code been removed? | • A-7, #5-7 |
| **3.6** | **Review the hardware/software (HW/SW) integration process data to determine compliance to DO-178B Table A-6.** | |
| 3.6.1 | How is each objective for the HW/SW integration process met? (Ask developer to explain how they meet objectives #1-5 of DO-178B Table A-6.) | • A-5, #7<br>• A-6, #1-5 |
| 3.6.2 | Does the HW/SW integration comply with the plans? | • A-5, #7<br>• A-6, #1-5 |
| 3.6.3 | If an RTOS is used, have RTOS integration issues been addressed, such as:<br>• Is there a process to address the RTOS integration? Has it been specified in the plan?<br>• Are the error codes of the RTOS specified? Where?<br>• Are the error codes used properly by the applicant?<br>• Do the application developer's standards specify how the RTOS (and users guide) is to be used?<br>• Are RTOS limitations (e.g., program constraints) documented in the user's development standards?<br>• Is the applicant using the RTOS as it was intended? | • A-6, #1-5 |

| Item # | SOI #3 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| **3.7** | **Determine if data and control coupling have been properly carried out. (See CAST-19 [7], Data and Control Coupling Clarification, for further information)** | • A-7, #8 |
| **3.8** | **Review verification cases and procedures.** | |
| 3.8.1 | Does each test case have inputs, conditions, and expected results? | • A-7, #3-8 |
| 3.8.2 | Does each test case have procedures for test set-up (to include environment), test execution, and pass-fail criteria? | • A-7, #1 |
| 3.8.3 | If test cases are run on a simulator or emulator (for DO-178B compliance), have any of the test steps been eliminated by the simulator or emulator? If so, has the simulator or emulator been qualified? | • A-1, #3<br>• A-8, #6 |
| **3.9** | **Review verification results.** | |
| 3.9.1 | Have all results of inspections, analyses, and tests been documented with "pass" or "fail"? | • A-7, #2 |
| 3.9.2 | Witness at least one requirements-based test and consider the following questions:<br>• Are the tests repeatable?<br>• Are the tests complete?<br>• Do the results agree with what was included in the test results?<br>• Does the test verify the requirement? | • A-7, #1-8 |
| 3.9.3 | Have all discrepancies between expected results and actual results been documented and explained? | • A-7, #2 |
| 3.9.4 | Does traceability exist between test cases and the high-level requirements? If appropriate, does traceability from requirements to the structural coverage objectives exist? | • A-7, #3 |
| 3.9.5 | Have the high-level requirements been tested/verified? | • A-7, #3 |
| 3.9.6 | Have low-level requirements been tested/verified? | • A-7, #4 |
| 3.9.7 | Is the structural test coverage correct and sufficient?<br>• Modified condition decision coverage (level A)<br>• Decision coverage (level A, B)<br>• Statement coverage (level A, B, C)<br>• Data coupling (level A, B, C)<br>• Control coupling (level A, B, C) | • A-7, #5-8 |
| 3.9.8 | Do test cases cover abnormal conditions and out-of-range data (robustness)? | • A-7, #3-8 |
| 3.9.9 | Does the reviewed data support the process defined in the Software Verification Plan? | • A-7, #1 |

| Item # | SOI #3 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| **3.10** | **Review the configuration management data to assess compliance to DO-178B Table A-8.** | |
| 3.10.1 | Review Configuration Identification and consider the following questions: | |
| 3.10.1.1 | Are all software life cycle data uniquely identified? | • A-8, #1 |
| 3.10.1.2 | Does configuration identification provide for the identification of the product, components, and sub-components – both individually and collectively? | • A-8, #1 |
| 3.10.1.3 | Does configuration identification of data occur prior to use of that data by another life cycle process? | • A-8, #1 |
| 3.10.2 | Review Baselining Activity and consider the following questions: | |
| 3.10.2.1 | Does the product baseline include the following data:<br>• PSAC, Configuration Index, and the Accomplishment Summary?<br>• Software requirements data?<br>• Each source code component?<br>• Previously developed software, if used in the software product?<br>• Instructions for building the executable code to include compiling and linking?<br>• Software life cycle (development) environment?<br>• Executable Object Code? | • A-8, #2 |
| 3.10.2.2 | Does development include the following in the product baseline:<br>• Software Development, Verification, Configuration Management and Quality Assurance Plans?<br>• Requirements, Design, and Code Standards?<br>• Design Description?<br>• Software Verification Cases and Procedures? | • A-8, #2 |
| **3.11** | **Review the Problem Reports and changes to software life cycle data for impact on software code.** | |
| 3.11.1 | Is the applicant documenting problems as described in their plans (e.g., problem reports)? | • A-8, #3 |
| 3.11.2 | Is there a problem reporting process in place? Is there a change control process? Is the applicant following the process? | • A-8, #3 |
| 3.11.3 | Does the problem report adequately describe the deficiency or anomalous behavior and the proposed change(s)? | • A-8, #3 |
| 3.11.4 | Are all effected software module(s) identified? | • A-8, #3 |
| 3.11.5 | Was the configuration updated to reflect the new version(s)? | • A-8, #3 |
| 3.11.6 | Does the description in the problem report adequately describe the change made? | • A-8, #3 |

| Item # | SOI #3 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 3.11.7 | Was the correct form used? | • A-8, #3 |
| 3.11.8 | How was change authorization confirmed? | • A-8, #3 |
| 3.11.9 | Was the change documented in the prologue header? | • A-8, #3 |
| 3.11.10 | If the change affected the software design was the design data updated or a change applied to the baseline? | • A-8, #3 |
| 3.11.11 | If the change effected the requirements, were the requirements updated or a change applied to the baseline? | • A-8, #3 |
| 3.11.12 | Were retest, regression analysis, and safety affects addressed in the change process? | • A-8, #3 |
| **3.12** | **Review the Archival, Retrieval, and Release Procedures.** | |
| 3.12.1 | Was the product protected from unauthorized changes? | • A-8, #4 |
| 3.12.2 | Does the storage medium minimize risk of deterioration and regeneration of errors? | • A-8, #4 |
| 3.12.3 | Are copies stored in physically separate archives for disaster recovery? | • A-8, #4 |
| 3.12.4 | Are there provisions to make and verify error free copies, including executable document name, version, and date; paragraph numbers; requirements identification; and results? | • A-8, #4 |
| 3.12.5 | Has the development environment been preserved for future changes to software? | • A-8, #4 |
| **3.13** | **Review the Software Quality Assurance Data to assure compliance to DO-178B Table A-9.** | |
| 3.13.1 | Was the quality assurance plan followed? | • A-9, #1 |
| 3.13.2 | Is there objective evidence that the transition criteria were satisfied prior to transitioning to the next phase(s) of development? | • A-9, #2 |
| 3.13.3 | Were any of the reviewed processes or products deficient? If so, examine the SQA records and speak with the SQA manager in order to:<br>• Determine why it was not found earlier.<br>• If the process in place was not effective enough to detect the problem, ask what corrective action will be taken.<br>• Assess possible causes of quality assurance function deficiencies (e.g., overworked staff, staff lacks knowledge). | • A-9, #1 |
| **3.14** | **Perform a build and load, using the applicant's approved instructions.** | |
| 3.14.1 | Are the instructions repeatable and easily understood? | • A-8, #5 |
| **3.15** | **If tool qualification is required, review tool qualification data, considering the following questions:** | |

| Item # | SOI #3 Evaluation Activity/Question | DO-178B objective(s) |
|---|---|---|
| 3.15.1 | Do the plans state which tools are being qualified and the rationale for qualification? (Note: This might be in the Plan for Software Aspects of Certification or a separate tool qualification plan for verification tools.) | • A-1, #4 and applicable objectives |
| 3.15.2 | Are the specific tool requirements documented? DO-178B, section 12.2.3.15 lists the typical information that should be included in the Tool Operational Requirements document. | • A-1, #4 and applicable objectives |
| 3.15.3 | Does the Tool Operational Requirements make known all of the tool's functions? | • A-1, #4 and applicable objectives |
| 3.15.4 | If a qualified tool is used for structural coverage, does the tool qualification data address whether the tool needs to instrument the code to perform the analysis? If the tool does need to instrument the code, has the effect of the instrumentation on the code been assessed? | • A-1, #4 and applicable objectives |
| 3.15.5 | If the tool measures coverage at the object code level, is additional analysis available to support the equivalence of coverage at the object and source code levels? (**Note:** See CAST-17 [8] for information on coverage at the object code level.) | • A-1, #4 and applicable objectives |
| 3.15.6 | Is the tool qualification analysis sufficient to discover errors in the tool and limitations of the tool's functions? | • A-1, #4 and applicable objectives |
| 3.15.7 | Does the tool qualification data address how tool deficiencies that are found while the tools are being used in a certification project should be handled? | • A-1, #4 and applicable objectives |
| 3.15.8 | Does the tool qualification data detail how changes to the tool will be evaluated and controlled? | • A-1, #4 and applicable objectives |
| 3.15.9 | Are procedures for using each tool documented? | • A-1, #4 and applicable objectives |
| 3.15.10 | Are limitations of the tool that may affect assessment of coverage clearly documented and addressed (e.g., the limitations discussed in chapter 4 of the MC/DC tutorial)? | • A-1, #4 and applicable objectives |

## *NOTES:*

## 3.4 Activities for Stage of Involvement #4 – FINAL REVIEW

***Purpose***

- Determine if final compliance to all of the DO-178B objectives has been achieved and all open items addressed/dispositioned.

- Assess the Software Configuration Index, Software Life Cycle Environment Configuration Index, Software Accomplishment Summary, and any other documents not previously reviewed.

***When to Perform***

When the software life cycle is completed, and the following items have been completed by the applicant (or when deemed necessary):

- Software conformity review has been performed.

- Software Accomplishment Summary and Configuration Indexes have been reviewed and are correct.

- Formal signature process has been completed.

***Data to Review Prior to the Review***

Reports from SOI #1, SOI #2, and/or SOI #3; open items from SOI #1, SOI #2, and/or SOI #3; all plans; Software Accomplishment Summary; and Configuration Index.

***Data to Review at the Review***

Software Life Cycle Environment Configuration Index; Software Configuration Index; Problem Reports; Software Accomplishment Summary; Designees' findings/observations from pre-review activities; and any data that had issues in previous reviews.

***Number of Days***

1-2 days

Note 1: If SOI #1, SOI #2, and SOI #3 have been performed, SOI #4 may only be an assessment of whether or not outstanding issues have been resolved (in this case the review might take only 1 day).

Note 2: If SOI #1, SOI #2, and SOI #3 were not performed, this review could take much longer.

***Evaluation Activities and Questions***

During SOI #4 the Software Accomplishment Summary and Software Configuration Index are evaluated. Additionally, open items from previous reviews are evaluated to ensure that all of the DO-178B objectives and project issues are addressed. The table below describes some typical activities to perform and questions to ask during SOI #4.

***Instructions***

- There are nine major evaluation activities for Stage of Involvement #4 - Final Review.

- Review the questions for each activity in relationship to its corresponding DO-178B objective and software criticality.

| Item # | SOI #4 Evaluation Activity/Question | DO-178B objective (s) |
|---|---|---|
| 4.1 | Were activities performed from SOI #1, SOI #2, or SOI #3 that were not previously completed or that were not found satisfactory and required changes completed? | • A-10, #3 |
| 4.2 | Is the Software Accomplishment Summary in accordance with DO-178B, 11.20? | • A-10, #3 |
| 4.3 | Is the Software Life Cycle Environment Configuration Index in accordance with DO-178B, 11.15? | • A-10, #3 |
| 4.4 | Is the Software Configuration Index in accordance with DO-178B, 11.16? | • A-10, #3 |
| 4.5 | Are all required software life cycle data completed and signed? | • A-10, #3 |
| 4.6 | In assessing problem reports, determine if the following have been properly analyzed and addressed:<br>• Are there any open problem reports that affect safety?<br>• Are there any open problem reports that affect operations?<br>• Have problem reports been adequately analyzed? | • A-10, #3 |
| 4.7 | Complete the Summary of Compliances/Findings/Observations table for all objectives for the appropriate software level. | • A-10, #3 |
| 4.8 | Does the system still satisfy the safety assessment objectives? Ensure that the planned safety objectives were actually addressed in the project implementation. | • A-1, #1-4<br>• A-10, #3 |
| 4.9 | Ensure that all applicable DO-178B objectives have been satisfied. | • A-10, #3 |
| 4.10 | Was the software conformity review performed on the "as-built" system? Does the SCI capture the "as-built" information? (Reference sections 8.1c and 8.3 of DO-178B.) | • A-9, #3 |

## PART 4 - *SUMMARIZING COMPLIANCES, FINDINGS, AND OBSERVATIONS FOR EACH DO-178B OBJECTIVE*

This section provides a way to tie the applicant's activities and compliance with the objectives of DO-178B and to document any findings and observations. For each objective, the review team evaluates what evidence exists to verify compliance. The "Summary of Compliances/Findings/Observations" form is included on the following pages. This form flags any issues found during a software review that may be considered certification issues. If the objective is not met, the reason for non-compliance should be stated in the form. This form provides a summary and tracking mechanism for discussions with the applicant.

While carrying out the activities/questions in Part 3 of the Job Aid, findings and observations are typically made. Findings should be mapped against the objectives of DO-178B using the Summary of Compliances/Findings/Observations form in this section. Observations may also be recorded in these tables, if desired; however, it should be clearly stated which are observations and which are findings. Additionally, the compliances to DO-178B objectives that were evaluated should be documented.

The Summary of Compliances/Findings/Observations form has five columns. The first two columns summarize the DO-178B objectives. "Anx Ref #" refers to the appropriate table in DO-178B Annex A (e.g., 1-1 refers to Table 1 and Objective 1 in DO-178B Annex A). The third column provides a place to record the compliances/findings/observations found during a review. The fourth column summarizes which levels for which the particular objective is applicable. The last column provides a tie back to the Job Aid Tables in Part 3 of this Job Aid.

The Summary of Compliances/Findings/Observations form is provided electronically so it may be included in the review report. Completion of these tables should be performed during and/or after the software review and should be included as part of the review report. By the end of a project, all DO-178B objectives should be assessed somehow, either through FAA or designee review.

The actions and issues should also be noted during the review, but will likely be summarized in a separate table, since they may not tie to specific DO-178B objectives.

A sample of a completed Summary of Compliances/Findings/Observations form is included in Section 2 of Supplement 3.

**NOTE:** This section provides one way to summarize review results. Appendix A provides an alternate approach that has been found effective as well. In both cases, it is important to evaluate all applicable DO-178B objectives and to document where compliance has been found and where findings and/or observations exist.

| Anx Ref # | Objective Summary (Numbers are DO-178B section references) | Summary of Compliances/ Findings/Observations | Applicabl e Level | Job Aid Ref |
|---|---|---|---|---|
| 1-1 | Software development and integral processes activities are defined. 4.1a, 4.3 | | A/B/C/D | 1.1, 1.3, 1.4, 1.6, 1.7, 1.8, 1.10, 1.11, 4.8 |
| 1-2 | Transition criteria, inter-relationships and sequencing among processes are defined. 4.1b, 4.3 | | A/B/C | 1.1, 1.3, 1.4, 1.6, 1.7, 1.8, 1.11, 4.8 |
| 1-3 | Software life cycle environment is defined. 4.1c | | A/B/C | 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.10, 1.11, 3.8, 4.8 |
| 1-4 | Additional considerations are addressed. 4.1d | | A/B/C/D | 1.1, 1.2, 1.3, 1.4, 1.7, 1.11, 2.4, 2.13, 3.15, 4.8 |
| 1-5 | Software development standards are defined. 4.1e | | A/B/C | 1.1, 1.3, 1.9, 1.10, 1.11, 2.13 |
| 1-6 | Software plans comply with this document. 4.1f, 4.6 | | A/B/C | 1.1, 1.3, 1.10, 1.11 |
| 1-7 | Software plans are coordinated. 4.1g, 4.6 | | A/B/C | 1.1, 1.3, 1.7, 1.11 |
| 2-1 | High-level requirements are developed. 5.1.1a | | A/B/C/D | 1.4, 1.6, 2.1, 2.6, 2.12, 2.15, 2.16 |
| 2-2 | Derived high-level requirements are defined. 5.1.1b | | A/B/C/D | 1.4, 2.1, 2.6, 2.12, 2.15, 2.16 |
| 2-3 | Software architecture is developed. 5.2.1a | | A/B/C/D | 1.4, 2.1, 2.6, 2.12, 2.15, 2.16 |
| 2-4 | Low-level requirements are developed. 5.2.1a | | A/B/C/D | 1.4, 2.1, 2.6, 2.12, 2.15, 2.16 |
| 2-5 | Derived low-level requirements are defined. 5.2.1b | | A/B/C/D | 1.4, 2.1, 2.6, 2.12, 2.15, 2.16 |
| 2-6 | Source Code is developed. 5.3.1a | | A/B/C/D | 1.4, 2.6, 2.12, 2.13, 2.15, 2.16 |
| 2-7 | Executable Object Code is produced and integrated in the target computer. 5.4.1a | | A/B/C/D | 1.4, 2.15, 2.16 |
| 3-1 | Software high-level requirements comply with system requirements. 6.3.1a | | A/B/C/D | 2.1, 2.15, 2.16 |
| 3-2 | High-level requirements are accurate and consistent. 6.3.1b | | A/B/C/D | 2.1, 2.15, 2.16 |

| Anx Ref # | Objective Summary (Numbers are DO-178B section references) | Summary of Compliances/ Findings/Observations | Applicabl e Level | Job Aid Ref |
|---|---|---|---|---|
| 3-3 | High-level requirements are compatible with target computer. 6.3.1c | | A/B | 2.1, 2.3, 2.5, 2.15, 2.16 |
| 3-4 | High-level requirements are verifiable. 6.3.1d | | A/B/C | 2.1, 2.6, 2.15, 2.16 |
| 3-5 | High-level requirements conform to standards. 6.3.1e | | A/B/C | 2.1, 2.15, 2.16 |
| 3-6 | High-level requirements are traceable to system requirements. 6.3.1f | | A/B/C/D | 2.1, 2.15, 2.16 |
| 3-7 | Algorithms are accurate. 6.3.1g | | A/B/C | 2.1, 2.15, 2.16 |
| 4-1 | Low-level requirements comply with high-level requirements. 6.3.15a | | A/B/C | 2.1, 2.2, 2.15, 2.16 |
| 4-2 | Low-level requirements are accurate and consistent. 6.3.15b | | A/B/C | 2.2, 2.15, 2.16 |
| 4-3 | Low-level requirements are compatible with target computer. 6.3.15c | | A/B | 2.2, 2.5, 2.15, 2.16 |
| 4-4 | Low-level requirements are verifiable. 6.3.15d | | A/B | 2.2, 2.6, 2.15, 2.16 |
| 4-5 | Low-level requirements conform to standards. 6.3.15e | | A/B/C | 2.2, 2.15, 2.16 |
| 4-6 | Low-level requirements are traceable to high-level requirements. 6.3.15f | | A/B/C | 2.1, 2.2, 2.15, 2.16 |
| 4-7 | Algorithms are accurate. 6.3.15g | | A/B/C | 2.2, 2.15, 2.16 |
| 4-8 | Software architecture is compatible with high-level requirements. 6.3.3a | | A/B/C | 2.2, 2.3, 2.15, 2.16 |
| 4-9 | Software architecture is consistent. 6.3.3b | | A/B/C | 2.2, 2.3, 2.15, 2.16 |
| 4-10 | Software architecture is compatible with target computer. 6.2.3c | | A/B | 2.2, 2.3, 2.5, 2.15, 2.16 |
| 4-11 | Software architecture is verifiable. 6.3.3d | | A/B | 2.2, 2.3, 2.6, 2.15, 2.16 |
| 4-12 | Software architecture conforms to standards. 6.3.3e | | A/B/C | 2.2, 2.3, 2.15, 2.16 |
| 4-13 | Software partitioning integrity is confirmed. 6.3.3f | | A/B/C/D | 2.2, 2.3, 2.6, 2.14, 2.15, 2.16 |
| 5-1 | Source Code complies with low-level requirements. 6.3.4a | | A/B/C | 2.4, 2.13, 2.15, 2.16 |

| Anx Ref # | Objective Summary (Numbers are DO-178B section references) | Summary of Compliances/ Findings/Observations | Applicable Level | Job Aid Ref |
|---|---|---|---|---|
| 5-2 | Source Code complies with software architecture. 6.3.4b | | A/B/C | 2.4, 2.13, 2.15, 2.16 |
| 5-3 | Source Code is verifiable. 6.3.4c | | A/B | 2.4, 2.13, 2.15, 2.16 |
| 5-4 | Source Code conforms to standards. 6.3.4d | | A/B/C | 2.4, 2.13, 2.15, 2.16 |
| 5-5 | Source Code is traceable to low-level requirements. 6.3.4e | | A/B/C | 2.4, 2.13, 2.15, 2.16 |
| 5-6 | Source Code is accurate and consistent. 6.3.4f | | A/B/C | 2.4, 2.13, 2.15, 2.16 |
| 5-7 | Output of software integration process is complete and correct. 6.3.5 | | A/B/C | 2.4, 2.13, 2.15, 2.16, 3.6 |
| 6-1 | Executable Object Code complies with high-level requirements. 6.4.2.1, 6.4.3 | | A/B/C/D | 2.11, 3.2, 3.6 |
| 6-2 | Executable Object Code is robust with high-level requirements. 6.4.2.2, 6.4.3 | | A/B/C/D | 2.11, 3.2, 3.6 |
| 6-3 | Executable Object Code complies with low-level requirements. 6.4.2.1, 6.4.3 | | A/B/C | 2.11, 3.2, 3.6 |
| 6-4 | Executable Object Code is robust with low-level requirements. 6.4.2.2, 6.4.3 | | A/B/C | 2.11, 3.2, 3.6 |
| 6-5 | Executable Object Code is compatible with target computer. 6.4.3a | | A/B/C/D | 2.11, 3.2, 3.6 |
| 7-1 | Test procedures are correct. 6,3,6b | | A/B/C | 3.3, 3.4, 3.8, 3.9 |
| 7-2 | Test results are correct and discrepancies explained. 6.3.6c | | A/B/C | 3.4, 3.5, 3.9 |
| 7-3 | Test coverage of high-level requirements is achieved. 6.4.4.1 | | A/B/C/D | 3.3, 3.5, 3.8, 3.9 |
| 7-4 | Test coverage of low-level requirements is achieved. 6.4.4.2 | | A/B/C | 3.3, 3.5, 3.8, 3.9 |
| 7-5 | Test coverage of software structure (modified condition/decision) is achieved. 6.4.4.2 | | A | 1.7, 3.3, 3.5, 3.8, 3.9 |

| Anx Ref # | Objective Summary (Numbers are DO-178B section references) | Summary of Compliances/ Findings/Observations | Applicable Level | Job Aid Ref |
|---|---|---|---|---|
| 7-6 | Test coverage of software structure (decision coverage) is achieved. 6.4.4.2a, 6.4.4.2b | | A/B | 1.7, 3.3, 3.5, 3.8, 3.9 |
| 7-7 | Test coverage of software structure (statement coverage) is achieved. 6.4.4.2a, 6.4.4.2b | | A/B/C | 1.7, 3.3, 3.5, 3.8, 3.9 |
| 7-8 | Test coverage of software structure (data coupling and control coupling) is achieved. 6.4.4.2c | | A/B/C | 1.7, 3.3, 3.7, 3.8, 3.9 |
| 8-1 | Configuration items are identified. 7.2.1 | | A/B/C/D | 1.5, 2.7.1, 3.10.1 |
| 8-2 | Baselines and traceability are established. 7.2.2 | | A/B/C/D | 1.5, 2.7.2, 3.10.2 |
| 8-3 | Problem reporting, change control, change review, and configuration status accounting are established. 7.2.3, 7.2.4, 7.2.5, 7.2.6 | | A/B/C/D | 1.5, 2.8, 3.11 |
| 8-4 | Archive, retrieval, and release are established. 7.2.7 | | A/B/C/D | 1.5, 2.9, 3.12 |
| 8-5 | Software load control is established. 7.2.8 | | A/B/C/D | 1.5, 3.14 |
| 8-6 | Software life cycle environment control is established. 7.2.9 | | A/B/C/D | 1.5, 2.7.3, 2.7.4, 3.8 |
| 9-1 | Assurance is obtained that software development and integral processes comply with approved software plans and standards. 8.1a | | A/B/C/D | 1.6, 2.1, 2.2, 2.10, 3.1, 3.13 |
| 9-2 | Assurance is obtained that transition criteria for the software life cycle processes are satisfied. 8.1b | | A/B | 1.6, 2.10, 3.13 |
| 9-3 | Software conformity review is conducted. 8.1c, 8.3 | | A/B/C/D | 1.6, 2.10, 4.10 |
| 10-1 | Communication and understanding between the applicant and the certification authority is established. 9.0 | | A/B/C/D | *1.1 - 1.11* |

| Anx Ref # | Objective Summary (Numbers are DO-178B section references) | Summary of Compliances/ Findings/Observations | Applicabl e Level | Job Aid Ref |
|---|---|---|---|---|
| 10-2 | The means of compliance is proposed and agreement with the Plan for Software Aspects of Certification is obtained. 9.1 | | A/B/C/D | 1.3 |
| 10-3 | Compliance substantiation is provided. 9.2 | | A/B/C/D | 4.1-4.9 |

# APPENDIX A – *ALTERNATE APPROACH FOR RECORDING FINDINGS/OBSERVATIONS/COMPLIANCES*

The table below provides another approach for documenting review compliances, findings, and observations. Each column in the table is described by the notes below.

| Item # | Doc | 178B Obj | F/O/C | Description | Applicant Response | Status |
|---|---|---|---|---|---|---|
| Note 1 | Note 2 | Note 3 | Note 4 | Note 5 | Note 6 | Note 7 |
| | | | | | | |
| | | | | | | |
| | | | | | | |

Note 1: Number the item for future reference.
Note 2: Include document or data that the finding or observation is made against.
Note 3: Include the applicable DO-178B objective(s)
Note 4: Classify the item as a finding (F), observation (O), or compliance (C). Issues (I), actions (A), and notes (N) might also be includes in this format.
Note 5: Describe the review finding or observation – be specific
Note 6: Allow space for the applicant to respond to each item of the review report. This will typically include their strategy for dealing with the finding or observation.
Note 7: Document status of items, as the project matures. The goal is for all findings to be closed or properly dispositioned.

# Notes:

# APPENDIX B – *REFERENCES*

[1] RTCA/DO-178B, *Software Considerations in Airborne Systems and Equipment Certification*, December 1, 1992. Available at: http://www.rtca.org

[2] Federal Aviation Administration, Order 8110.49, *Software Approval Guidelines*, June 3, 2003. Available at: http://www2.faa.gov/certification/aircraft/av-info/software/software.htm.

[3] K. Hayhurst, D. Veerhusen, J. Chilenski, and L. Rierson, *A Practical Tutorial on Modified Condition/Decision Coverage*, NASA Report: NASA/TM-2001-210876, May 2001. Available at: http://av-info.faa.gov/software

[4] CAST-12, *Guidelines for Approving Source Code to Object Code Traceability*, December 2002. Available at: http://www2.faa.gov/certification/aircraft/av-info/software/software.htm.

[5] CAST-8, *Use of the C++ Programming Language*, January 2002. Available at: http://www2.faa.gov/certification/aircraft/av-info/software/software.htm.

[6] CAST-18, *Reverse Engineering in Certification Projects*, 2003. Available at: http://www2.faa.gov/certification/aircraft/av-info/software/software.htm.

[7] CAST-19, *Data and Control Coupling Clarification*, 2003. Available at: http://www2.faa.gov/certification/aircraft/av-info/software/software.htm.

[8] CAST-17, *Structural Coverage of Object Code*, 2003. Available at: http://www2.faa.gov/certification/aircraft/av-info/software/software.htm.

[9] FAA software web-site: http://www2.faa.gov/certification/aircraft/av-info/software/software.htm.