U.S. Department
of Transportation
**Federal Aviation
Administration**

# Advisory Circular

**AC 20-RSC**

# REUSABLE SOFTWARE COMPONENTS

# DRAFT 8.3 –6/11/03

**Date: XXXXXXXX**

**TABLE OF CONTENTS**

**SECTION** **PAGE**

**1. PURPOSE. ..... 1**

**2. BACKGROUND. ..... 1**

**3. RELATED DOCUMENTS. ..... 2**

**4. DOCUMENT OVERVIEW. ..... 2**

**5. DISCUSSION. ..... 3**

**6. GUIDELINES FOR THE RSC DEVELOPER. ..... 4**

**7. DATA SUPPLIED TO THE RSC INTEGRATOR AND/OR APPLICANT. ..... 6**

**8. GUIDELINES FOR THE INTEGRATOR AND APPLICANT USING THE RSC. ..... 7**

**9. EXPECTATIONS FROM CERTIFICATION AUTHORITIES ON THE FIRST USE OF AN RSC. ..... 9**

**10. THE RSC ACCEPTANCE LETTER. ..... 9**

**11. EXPECTATIONS FROM CERTIFICATION AUTHORITIES ON THE SUBSEQUENT USE OF AN ACCEPTED RSC. ..... 10**

**12. COMMON SOFTWARE REUSE ISSUES. ..... 11**

**13. CHANGES TO REUSABLE SOFTWARE COMPONENTS. ..... 13**

**14. CONCURRENT USE OF AN RSC. ..... 17**

**APPENDIX 1 – DEFINITIONS OF TERMS ..... 1**

**APPENDIX 2 - ACRONYMS ..... 1**

**APPENDIX 3 – SAMPLE FORMAT FOR RSC DEVELOPER'S TABLE ..... 1**

DRAFT

U.S. Department of Transportation
**Federal Aviation Administration**

# Advisory Circular

| | | |
|---|---|---|
| **Subject:** **REUSABLE SOFTWARE COMPONENTS** | **Date:** XXXXXXXX | **AC No:** 20-RSC |
| | **Initiated By:** AIR-120 | **Change:** |

## 1. PURPOSE.

**a.** This advisory circular (AC) shows one acceptable way, but not the only way, for reusable software component (RSC) developers, integrators, and applicants to gain Federal Aviation Administration's (FAA) acceptance of a software component that may makeup a part of a system's software application. Like all advisory material, this AC is not mandatory and does not constitute a regulation. Because the method of compliance presented in this AC is not mandatory, the term "must" used herein applies only to an applicant who chooses to follow the method prescribed in this AC. This AC also shows a method to get credit for the reuse of component in follow-on projects, including receiving "credit" for full or partial compliance to the objectives in Annex A of RTCA/DO-178B. When this AC is followed and if no safety concerns are apparent, the FAA will grant acceptance for the RSC by writing an acceptance letter. If the RSC is unchanged and meets the limitations stated in the RSC acceptance letter, it may be reused without additional FAA review of the RSC data. This AC requires that the RSC being considered for acceptance have its own set of software life cycle data**.**

**b.** This AC applies to the approval of airborne systems and equipment and the software aspects of those systems related to type certificates (TC), supplemental type certificates (STC), amended supplemental type certificates (ASTC), amended type certificates (ATC), and Technical Standard Order (TSO) authorizations. For TSO authorized articles, the RSC acceptance letter will typically not be granted until the TSO authorized article and the RSC have received installation approval as part of a TC, STC, ASTC, or ATC. This practice is necessary because of the highly integrated and complex nature of software in airborne systems and equipment.

**2. BACKGROUND.** Economic incentives and technical advances in software development have made it desirable to develop a RSC that can be integrated into a number of target computers, as determined by the integrator and/or applicant. In these cases, a

DRAFT

developer of an RSC may partially meet the applicable RTCA/DO-178B objectives, while the integrator and/or applicant are responsible for completing the software and certification compliance activities. Examples of potential RSCs include software libraries, input and output data files, operating systems, and communication protocols.

## 3. RELATED DOCUMENTS.

**a. Code of Federal Regulations (CFR).** Title 14 of the Code of Federal Regulations parts 21, 23, 25, 27, 29, 33, and 35 are applicable to this AC. Copies of the CFRs are available from the FAA website at http://www.airweb.faa.gov/rgl.

**b. FAA Advisory Circulars**. AC 20-115B, RTCA, Inc., Document RTCA/DO-178B, dated January 11, 1993, offers a way to get FAA approval of software. The intent of this RSC AC is to provide guidance for applicants that use RTCA/DO-178B as their means of showing compliance for software components. This AC supplements RTCA/DO-178B and AC 20-115B, for approving some or all of the RTCA/DO-178B objectives for individual components of a system's software application. If an applicant proposes a means of compliance other than RTCA/DO-178B, the FAA will decide whether this AC applies and whether additional policy or guidance is warranted. You can obtain copies of this AC, AC 20-115B, and other ACs from the FAA website at http://www.airweb.faa.gov/rgl.

**c. FAA Policy Documents.** Federal Aviation Administration (FAA) Order 8110.4, Type Certification (as amended) is relevant to this AC. Copies of Order 8110.4 and other orders are available from the FAA website at http://www.airweb.faa.gov/rgl.

**d. RTCA, Inc. Documents.** You may purchase copies of RTCA documents from RTCA, Inc., 1828 L Street, NW, Suite 805, Washington, D.C. 20036. Alternatively, copies may be purchased on-line at http://www.rtca.org/. RTCA documents referenced in this AC are:

**(1)** RTCA, Inc., Document RTCA/DO-178B, Software Considerations in Airborne Systems and Equipment Certification, dated December 1, 1992.

**(2)** RTCA, Inc., Document RTCA/DO-248B, Final Report for Clarification of DO-178B Software Considerations in Airborne Systems and Equipment Certification, dated October 12, 2001.

## 4. DOCUMENT OVERVIEW. The following sections are included in this AC:

**a.** Sections 1 through 5 set the stage for the AC by providing background and introductory information.

**b.** Sections 6 through 8 provide guidelines for the RSC developers, integrators, and applicants, when developing or using an RSC.

DRAFT

**c.** Sections 9 through 10 provide typical activities that the RSC developers, integrators, and applicants can expect from the certification authorities for the first acceptance of an RSC and its subsequent use.

**d.** Section 11 discusses the expectations from the certification authorities on the subsequent use of an accepted RSC.

**e.** Section 12 discusses common issues that should be addressed when developing and using RSCs. These issues may affect multiple RTCA/DO-178B objectives. Section 12 does not present an exhaustive list of issues that may arise, since each project will have its own specific issues.

**f.** Section 13 addresses changes to an RSC.

**g.** Section 14 considers concurrent uses of an RSC.

**h.** Appendix 1, Definitions of Terms, defines the terms used in this AC. **Please review this appendix prior to reading the AC in order establish a consistent terminology.**

**i.** Appendix 2, Acronyms, lists the acronyms used in this AC.

**j.** Appendix 3, Sample Format for RSC Developer's Table, provides a sample format of a RSC developer's table.

## 5. DISCUSSION.

**a.** The first acceptance of a reusable software component must be performed during an actual project (such as, a TC, ATC, STC, ASTC, or TSO authorization project). This may require extra resources from software component developers, the integrator, the applicant, and the certification authority. Subsequent acceptance of the RSC for a different project will likely require less effort and resources, when the guidelines in this AC are followed.

**b.** This reuse guidance applies only when all the stakeholders (the applicant, integrator, RSC developer, and certification authority) agree that the software component is reusable. The RSC Plan for Software Aspects of Certification (PSAC) and the system-level PSAC of the first applicant are the recommended vehicles for agreement and communication among stakeholders. Agreeing on the reuse concept is important because the first applicant will likely use additional resources to qualify the component as reusable. When there is no agreement, then the traditional approach to software development and approval should be followed (see Section 5 of this AC).

**c.** Each RSC developer's project will have different limitations, needs, and issues. For example, one developer may package a project so it fully meets a particular objective in Annex A of RTCA/DO-178B. Another developer may only partially meet that same objective. This may be due to some project-specific issues, or additional coordination with the integrator to augment the work of the RSC developer. Sections 5 through 7 of this AC

guide the RSC developer, integrator, and applicant. The guidelines are meant to be flexible enough to satisfy the multiple needs of the RSC developer, integrator, and applicant. However, the guidelines are also detailed enough to ensure that relevant certification issues are addressed.

**d.** Applicants are primarily responsible for submitting compliance data, communicating with certification authorities, and following the certification liaison process for the project. However, there may be some communication between the certification authority and the RSC developer for the reuse aspects of the program.

**6. GUIDELINES FOR THE RSC DEVELOPER.** Currently, there are no procedures for RSC developers to directly transfer their accepted data from one project to the next and across company boundaries. Traditionally, RSC developers provided substantiation in one of two ways. First, by resubmitting the data package and repeating the work for each system's application. Second, by providing traceability through the TC, ATC, STC, ASTC, or TSO approval back to the desired data and defending the validity of the original approval basis for each application This AC addresses the reuse of software components and software life cycle data across company boundaries. The RSC developer must do the following:

**a.** Produce a PSAC for the RSC as early as possible in the project. The PSAC must include the information discussed in Section 11.1 of RTCA/DO-178B and detail the RSC developer's plans for satisfying each applicable RTCA/DO-178B objective. The PSAC must also identify which objective(s) will not be satisfied, and which objective(s) will be partially satisfied. Additionally, the PSAC must explicitly state the RSC developer's agreement that the RSC is being developed with the intent to reuse it in future projects.

**b.** Consider and address, as applicable, the common reuse issues documented in Section 11 of this AC.

**c.** For each RTCA/DO-178B objective applicable to the software level, document the information listed below (either directly in the PSAC or by reference to other documents) with sufficient detail for certification authority concurrence and integrator and/or applicant usage of the RSC. The RSC developer may include this information in a table with columns for the objective number, objective description, credit being sought, assumptions, means of compliance, and remaining activities (see a sample format in Appendix 3). Since target computer-specific and system-specific issues may be uncertain early in the project, the PSAC may list preliminary information that will be updated in PSAC revisions and/or the Software Accomplishment Summary (SAS). Some reuse details may not be finalized until the end of the project. The following information must be thoroughly documented for review by certification authorities and for usage by integrators and/or applicants:

**(1) Credit being sought for the objective.** The PSAC or referenced document must specify if full, partial, or no credit is being sought for the objective.

**(2) Assumptions of the RSC developer on the behavior of the RSC users.** Provide sufficient justification to ensure that the original acceptance is valid if the assumptions are satisfied.

**(3) Means of compliance for the objective.** The PSAC must document what data supports compliance to each objective (document titles and/or a description of the type of data to be provided for compliance).

**(4) Activities remaining for the integrator and/or applicant.** The PSAC must document what an applicant and/or integrator need to do to fully satisfy any partial or unsatisfied objective.

**d.** Document the following safety-related items in the PSAC and SAS:

**(1)** The software level for the RSC,

**(2)** Any safety objectives or safety-related requirements allocated to the RSC,

**(3)** Known effects of the RSC to the safety assessment,

**(4)** Architectural and design features supporting any portion of the safety analysis, partitioning, or other protection strategies, and

**(5)** Any safety assumptions that support the use of the RSC.

**e.** Additionally, the RSC developer must also produce a safety assessment based on the intended RSC usage in order to document any known issues that could affect safety.

**f.** Obtain agreement by all stakeholders for the first application by coordinating the PSAC and any other plans (e.g., Software Development Plan (SDP), Software Verification Plan (SVP), Software Quality Assurance Plan (SQAP), and Software Configuration Management Plan (SCMP)) with the certification authority, designees (if delegated), and the applicant and/or integrator.

**g.** Develop the RSC in compliance with the agreed upon plans. As previously stated, the RSC developer must produce the RTCA/DO-178B software life cycle data and documentation identified in this AC for the RSC (such as, plans, standards, requirements, quality assurance records, and configuration management records).

**h.** Inform the certification authority, designees (if delegated), integrator, and applicant of both development progress and any deviations from plans, to allow for timely reviews and adjustments as necessary.

**i.** Submit the Software Configuration Index (SCI) and the SAS to the certification authority through the project integrator, applicant, or designee (if delegated), when

completed. The SAS must include or refer to the software life cycle data of RTCA/DO-178B, Section 11, and the information discussed in Sections 5a through 5e of this AC.

**7. DATA SUPPLIED TO THE RSC INTEGRATOR AND/OR APPLICANT.** The RSC developer must supply the appropriate software life cycle data to the integrator and/or applicant to support certification of the systems using the RSC. Typically, the RSC developer supplies the following data to the RSC integrator and/or applicant (except for item i below):

**a.** The type design data listed in Section 9.4 of RTCA/DO-178B for the RSC (that is, Software Requirements Data, Design Description, Source Code, Executable Object Code, SCI, and SAS).

**b.** The PSAC, which lists the credit sought for each RTCA/DO-178B objective.

**c.** Interface information (for example, interface control document and porting guide). Include any hardware and software resource requirements (such as, timing and memory).

**d.** Installation or integration procedures and limitations, sufficient to ensure that the RSC will be properly used, integrated, and installed. They must be detailed enough to identify unique aspects of the installation or integration. The limitations and procedures must include, as a minimum:

**(1)** Equipment specifications required for proper operation and performance of the RSC.

**(2)** A list of any RSC sub-components (by part number and version number).

**(3)** Instructions for periodic maintenance and calibration needed for continued airworthiness once the software is installed.

**e.** Data to support the integrator and/or applicant's completion of partially satisfied or unsatisfied objectives. As an example, if partial credit is sought for objective 1 of RTCA/DO-178B Table A-1 (Software development and integral processes are defined), clearly define to the integrator and/or applicant what that partial credit entails and what they need to do to complete the credit for the installation. The necessary data to support that "partial" credit must also be made available to the integrator and/or applicant.

**f.** Test cases and procedures to be re-executed on the target environment. This should include a list of test cases and procedures affected by any integrator and/or applicant's settable parameters. The integrator and/or applicant should consider the total requirements for system and sub-system testing:

**(1)** applicable credit for reusable tests,

**(2)** re-test where new settings affect the requirements and code, and

**(3)** creation of new test cases and procedures to complete all test objectives.

**g.** A summary of open problem reports on the RSC and analysis of the operational and safety effects. Document this information in the RSC SAS, and if the information is known at the beginning of the project, include it in the RSC PSAC.

**h.** The RSC developer must develop a data sheet for the RSC. This data sheet must summarize RSC functions, limitations, safety concerns, assumptions, configuration, supporting data, and other relevant information in a concise manner that supports the integrator and/or applicant's use of the RSC. The data sheet will be attached to the FAA acceptance letter.

**i.** The following data-related items must also be considered by the RSC developers (although they may not result in submittals):

**(1)** Any RTCA/DO-178B software life cycle data not listed above, but used in the software development and approval process, must be made available to the applicant, integrator, and certification authority (for example, Software Quality Assurance (SQA) and Software Configuration Management (SCM) records).

**(2)** Irrespective of any legal and proprietary issues and agreements about the delivery of software life cycle data between the applicant and the RSC developer, the data must be available to the certification authorities at all times for their review and inspection. A process may be set up to make the data available to the applicant without actually supplying the data to the applicant (for example, a data/software escrow). This data must be accessible to the certification authority to determine compliance, or in the event of safety problems with the target system (see 14 CFR §§ 21.277). The data may also need to be available to the applicant, if the target system or RSC requires modification (reference 14 CFR § 21.301 through § 21.305, and FAA Order 8110.4).

**(3)** Data needed to support changes to the RSC must be identified. For example, if the developer goes out of business, this data will help support continued airworthiness. Title 14 of the Code of Federal Regulations, Part 21, Certification Procedures for Products and Parts, is supplemented by FAA Order 8110.4, Type Certification (Chapters 2 and 3), provides guidance for the issuance and preservation of type design data for maintaining the continued airworthiness of products.

**(4)** The RSC developer must keep a list of all aviation customers buying or using their components to support continued airworthiness across multiple products. It is also recommended that the RSC developer and users set up a process to share problem reports in support of users required to comply with 14 CFR ⇓ 21.3.

**8. <u>GUIDELINES FOR THE INTEGRATOR AND APPLICANT USING THE RSC</u>.** Sometimes the integrator and applicant are the same company or organization and sometimes they are separate entities. The guidelines for the integrator and applicant are

listed below. The applicant is responsible for ensuring that these items are completed, even if some of the tasks are actually performed by an integrator. The applicant and integrator must:

**a.** Integrate the RSC developer's plans, documentation, limitations, compliance statement, mapping to RTCA/DO-178B objectives, software approval approach, and other relevant information into their own data submittals. If the integration is a subsequent reuse of an accepted RSC, consider the RSC acceptance letter and data sheet.

**b.** Specify the RTCA/DO-178B software life cycle data needed from the RSC developer that supports their project and continued airworthiness. This data is listed in Section 6 of this AC.

**c.** Produce a PSAC (and/or equivalent system-level certification plan) for the target system, including the information outlined in RTCA/DO-178B, Section 11.1. The PSAC must include the integrator and/or applicant's plans to address compliance with all RTCA/DO-178B objectives for the RSC and other software components of the target system. Additionally, the PSAC must explicitly state the agreement that the RSC will be developed with the intent to be reusable in other projects.

**d.** Produce other software plans (such as, SDP, SCMP, SVP, and SQAP) for their target system. Each plan must address the RSC integration and other software components used. For example, the SVP must cover the overall software verification program, plus any verification required to integrate the RSC and other components, and the credit proposed for the RSC developer's verification.

**e.** Consider the safety issues identified in the RSC developer's PSAC, SAS, and safety assessment; determine the applicability of those safety issues to the specific application; determine any additional safety issues for the specific application; and address all safety issues identified.

**f.** Coordinate the PSAC and other plans (as needed) with the certification authority and designees (if delegated) to get agreement on the project.

**g.** Follow the approved plans.

**h.** Analyze any open problem reports on the RSC to ensure that there are no safety or operational effects from the RSC in the specific application.

**i.** Validate the assumptions and claims for credit made by the RSC developer.

**j.** Validate and verify the throughput, timing, memory usage, resource usage, and other resource items of the RSC and other installed software components for the specific target environment.

DRAFT

**k**. Keep the certification authority and designees (if applicable) informed of the project status and approved plan deviations. This communication supports timely reviews by the certification authority and/or designees (if applicable) and approval of changed plans.

**l.** Submit SCI, SAS, and any other software life cycle data required to the certification authority. The SAS must include the information described in Section 11.20 of RTCA/DO-178B for the system's software. The SCI and SAS must identify that the RSC has been included in the applicant's project, must identify the configuration of the RSC, and must identify the documentation configuration to support the RSC and other software components used in the system. Additionally, the SAS must include a description of how RTCA/DO-178B objectives that were not fully met by the RSC developer have been completely satisfied by the integrator and/or applicant.

**9. EXPECTATIONS FROM CERTIFICATION AUTHORITIES ON THE FIRST USE OF AN RSC.** The RSC developer, integrator, and applicant should work closely with the certification authority throughout the RSC development and integration. To gain acceptance of an RSC in its first installation, the certification authority will typically:

**a.** Work closely with the applicant, integrator, and RSC developer to ensure that they followed Sections 5 and 7 of this AC.

**b.** Involve directorate personnel, headquarters personnel, technical specialists, and chief scientific and technical advisors (CSTAs), as needed, to address policy and technical issues in the project.

**c.** Review the RSC developer, applicant, and/or integrator's plans to ensure that the RTCA/DO-178B objectives will be satisfied.

**d.** Perform on-site or desk reviews of the software life cycle data and the capability of the RSC developer, applicant, and integrator, as needed, to ensure compliance to the RTCA/DO-178B objectives.

**e.** Approve data from the applicant, integrator, and RSC developer (as in a typical software program) for the target system software, when they satisfactorily complete their development and compliance activities.

**10. THE RSC ACCEPTANCE LETTER.** Upon successful certification of the product or authorization of the equipment using the RSC, the certification authority will write an acceptance letter for the RSC and will submit it to the RSC developer. A copy will be provided to the applicant and/or integrator. This letter documents the initial acceptance of the RSC and its suitability for use by other applicants and/or integrators. The acceptance letter typically includes:

**a.** The RSC document numbers and revision levels approved (for example, the SCI number and revision; the SAS number and revision; and any additional configuration

DRAFT

information not included in the SCI), and a general description of the RSC functionality and target environment.

**b.** The RSC developer's name and contact information.

**c.** The original applicant's and/or integrator's name and contact information using the RSC, the target computer system, the target computer environment, and other relevant information pertaining to the initial acceptance of the RSC.

**d.** Assumptions made by the RSC developer during the acceptance, including a reference to the RSC developer's SAS. This must include assumptions for each applicable RTCA/DO-178B objective. The assumptions must be sufficiently detailed that other certification authorities, RSC integrators, and applicants could apply the information to subsequent projects.

**e.** Summary of technical or policy issues during the initial acceptance and how those issues were addressed.

**f.** Summary of extra activities performed by the integrator and applicant to assure the RSC for the target system.

**g.** Contact information for the certification office that will address future questions about the RSC acceptance and subsequent reuse.

**h.** Software level of the RSC, any RSC limitations, and known safety or operational issues of the RSC.

**i.** RSC data sheet. A copy of the RSC data sheet is typically attached to the acceptance letter.

**11. EXPECTATIONS FROM CERTIFICATION AUTHORITIES ON THE SUBSEQUENT USE OF AN ACCEPTED RSC.** When a previously accepted RSC is submitted for subsequent reuse, the certification authority or the designee (if delegated) will:

**a.** Review the certification authority letter to the RSC developer documenting the initial acceptance. This letter may be obtained from the RSC developer or the certification authority that originally issued the acceptance.

**b.** Contact the certification office that performed the first acceptance (as documented in the acceptance letter) to discuss project details and to address any questions or concerns.

**c.** Work closely with the RSC applicant and integrator to ensure that they follow this AC's guidelines and address any additional certification issues.

**d.** Involve FAA Directorate personnel, headquarters personnel, technical specialists, and/or CSTAs, as needed, to address policy and technical issues in the project (particularly for Level 5A and B software).

**e.** Review the integrator and/or applicant's plans to ensure: (a) the objectives of RTCA/DO-178B will be satisfied, and (b) the assumptions and requirements documented for the RSC and for other software components used in the target system will be satisfied.

**f.** Perform on-site and desk reviews of the integrator and/or applicant's data and organizations capability (as needed). This ensures compliance to the RTCA/DO-178B objectives and approved plans, and compliance with the assumptions and requirements documented for the RSC and other software components.

**g.** Accept the applicant and/or integrator's data for the overall system software after they satisfactorily complete the integration and compliance activities.

**h.** Inform the original certification authority of the subsequent software acceptance, and report any issues that arose during the acceptance.

**12. <u>COMMON SOFTWARE REUSE ISSUES.</u>** There are several issues that may affect the reuse of software components. Some of the most common issues are discussed below (this is not an exhaustive list):

**a. Requirements definition.**

**(1)** RTCA/DO-178B discusses several types of requirements, including system requirements, safety-related requirements, high-level requirements, low-level requirements, and derived requirements. The RTCA/DO-178B discussion and objectives regarding requirements were developed with a traditional federated system in mind. In the traditional case, a single manufacturer is typically responsible for the software development and integration. When RSCs and multiple stakeholders become involved in the software development process, determining the levels of requirements become more difficult. Therefore, satisfying the RTCA/DO-178B objectives related to requirements require special attention.

**(2)** Each RSC developer must establish a plan to address the RTCA/DO-178B objectives related to high-level, low-level, and derived requirements. This plan must be clearly documented in the RSC PSAC and adhered to by the integrator and/or applicant. The plan should also be coordinated with the appropriate certification authorities as early in the program as possible.

**b. Re-verification.**

**(1)** When an RSC is reused, the question of how much re-verification needs to be performed often arises. Re-verification activities depend on the specific situation (such as, same or different processor, same or different compiler, same or different compiler options,

etc.). The RSC developer should document their overall verification (and re-verification) plans in the RSC PSAC. Additional details will likely be provided in the RSC SVP; however, the PSAC should have sufficient detail for the certification authority to determine that the approach will address the RTCA/DO-178B objectives.

**(2)** Some common re-verification questions to be considered are:

- How much re-verification is required if a different compiler type or optimization is used?
- How much re-verification is required if a different target environment is used?
- How much structural coverage analysis should be repeated if the target system changes?
- How much re-verification is needed for run-time and compiler libraries?
- If a new target environment is used, what kind of resource issues exist (for example, are there non-deterministic, dynamic memory allocation algorithms with the RSC that could create resource (such as, memory and execution time) issues in the new target environment?). If resource issues do exist, how will re-verification be carried out?

**c. Interface.** The RSC developer must provide interface data. This data must explicitly define what activities are required by the integrator and/or applicant to ensure that the RSC will function according to its requirements. Typical items included in interface data are:

- configuration parameters
- restrictions on tools
- additional verification activities
- memory and timing requirements
- external resources required by the RSC for proper functioning and performance
- definition of the communication mechanisms between the RSC and other software programs and the communication protocols with hardware components
- accessible variables and their characteristics
- variables and data required from the system and their characteristics (for example, inputs to RSC)
- bus and input/output ports and devices
- access mechanisms

**d. Partitioning and protection.** Although partitioning and protection will most likely be a function at the system level, the RSC itself may need some partitioning and protection. For example, there may be some maintenance code that is at a different software level than the operational flight program for the RSC. In some cases, the RSC might have specific protocols that facilitate protection and partitioning. These should be documented and evaluated by the integrator, applicant, and certification authorities.

**e. Coupling and cohesion.** Data and control coupling and cohesion issues between the RSC and other integrated software and hardware components must be carefully addressed. For example, modification to internal RSC data and structures should be prohibited or tightly controlled. If modifications are made to the RSC, then the guidance in Section 12 of this AC applies.

**f. Using qualified tools.** If qualified tools are used to develop and/or verify the RSC, reuse of those tools must be considered during the RSC development and acceptance. RTCA/DO-178B, Section 12.2 provides additional information on the tool qualification process and the supporting documentation.

**(1)** When qualified tools are used for the development and/or verification of an RSC, the Tool Qualification Plan and the Tool Accomplishment Summary (or PSAC and SAS for verification tools) must document any portions of the tool qualification that are to be completed by the applicant. For example, test procedures and cases might have some target dependencies and additional verification must be performed by the integrator and/or applicant.

> **NOTE:** Some developers have found that packaging the qualification data for each tool helps with reuse. For example, each verification tool used with an RSC might have its own Tool Qualification Plan, Tool Operational Requirements, and Tool Accomplishment Summary.

**(2)** The following tool qualification data must be provided to the applicant for all tools used with the RSC to be qualified:

**(a)** All tool plans,

**(b)** Tool Operational Requirements, and

**(c)** The Tool Accomplishment Summary to the applicant.

**(3)** All tool data not listed in Section 11.f.(2) of this AC must be available for review by the applicant and certification authority, as needed, to support continued airworthiness.

**g. Deactivated code.** Any information about deactivated code and the associated deactivation mechanisms must be identified by the integrator and/or applicant. Since the RSC may have many features to satisfy a broad audience, an approach to tailor the RSC to the specified requirements of an applicant's application is typically needed. This could result in sections of deactivated code that must be addressed as part of the overall software approval process.

## 13. CHANGES TO REUSABLE SOFTWARE COMPONENTS.

DRAFT

**a.** RSCs will likely change at some point in time. When an RSC is changed, the original reuse status no longer applies to the changed component. If the stakeholders desire to change a previously accepted RSC, the software component must be modified using the guidelines of this AC and re-accepted as part of an actual project.

**b.** When a RSC is changed, a change impact analysis must be performed to identify the changed and affected aspects of the software. The change impact analysis should follow a defined process to determine the potential impact of the change on continued operational safety of the aircraft. The following items should be addressed by the change impact analysis, as applicable:

**(1) Traceability analysis** identifies areas that could be affected by the software change. This includes the analysis of affected requirements, design, architecture, code, testing and analyses, as described below:

**(a) Requirements and design analysis** identifies the software requirements, software architecture, and safety-related software requirements impacted by the change. Additionally, the analysis identifies any additional features and/or functions being implemented in the system, assures that added functions are appropriately verified, and assures that the added functions do not adversely impact existing functions.

**(b) Code analysis** identifies the software components and interfaces impacted by the change.

**(c) Test procedures and cases analysis** identifies specific test procedures and cases that will need to be reexecuted to verify the changes, identifies and develops new or modified test procedures and cases (for added functionality or previously deficient testing), and assures that there are no adverse effects as a result of the changes. The absence of adverse effects may be verified by conducting regression testing at the appropriate hierarchical levels (such as aircraft flight tests, aircraft ground tests, laboratory system integration tests, simulator tests, bench tests, hardware/software integration tests, software integration tests, and module tests), as appropriate for the software level(s) of the changed software.

**(2) Memory margin analysis** assures that memory allocation requirements and acceptable margins are maintained.

**(3) Timing margin analysis** assures that the timing requirements, central processing unit task scheduling requirements, system resource contention characteristics, interface timing requirements, and acceptable timing margins are maintained.

**(4) Data flow analysis** identifies changes to data flow and coupling between components and assures that there are no adverse impacts.

**(5) Control flow analysis** identifies changes to the control flow and coupling of components and assures that there are no adverse impacts.

DRAFT

**(6) Input/output analysis** assures that the change(s) have not adversely impacted the input and output (including bus loading, memory access, and hardware input and output device interfaces) requirements of the product.

**(7) Development environment and process analyses** identify any change(s), which may adversely impact the software application or product (for example, compiler options or versions and optimization change; linker, assembler, and loader instructions or options change; or software tool change).

**(8) Operational characteristics analysis** evaluates that changes (such as changes to gains, filters, limits, data validation, interrupt and exception handling, and fault mitigation) do not result in adverse effects.

**(9) Certification maintenance requirements (CMR) analysis** determines whether new or changed CMRs are necessitated by the software change.

**(10) Partitioning analysis** assures that the changes do not impact any protective mechanisms incorporated in the design.

**NOTE:** The above list is not all-inclusive and depends on the product for which the modification is being made.

**c.** The change impact analysis should determine whether the change to the RSC could adversely affect safe operation of the system or product. The following are examples of areas that could have an adverse impact on safety or operation:

**(1) Safety-related information is changed.** For example:

**(a)** Previous hazards, identified by the system safety assessment, are changed.

**(b)** Failure condition categories, identified by the system safety assessment, are changed.

**(c)** Software levels are changed, particularly if the new software level is higher than the previous level.

**(d)** Safety-related requirements, identified by the system safety assessment, are changed.

**(e)** Safety margins are reduced.

**(2) Changes to operational or procedural characteristics of the aircraft that could adversely affect flight safety.** For example:

**(a)** Aircraft operational or airworthiness characteristics are changed.

**(b)** Flight crew procedures are changed.

**(c)** Pilot workload is increased.

**(d)** Situational awareness, warnings, and alerts are changed.

**(e)** Displayed information to make flight decisions is changed.

**(f)** Assembly and installation requirements are changed.

**(g)** Equipment interchangeability and/or interoperability with other equipment is changed.

**(h)** CMRs are changed or added.

**(3) New functions or features are added to the existing system functions that could adversely impact flight safety.**

**(4) Processors, interfaces, and other hardware components or the environment are changed in such a way that safety could be adversely affected** (see RTCA/DO-178B, Section 12.1.3).

**(5) Software life cycle data (requirements, code, and architecture) is significantly changed in such a way that it could adversely affect safety.** For example:

**(a)** Changes to software requirements, design, architecture, and code components (especially those affecting safety-related functions, partitioning, redundancy or safety monitors).

**(b)** Changes to code (source, object, and executable object) components that perform a safety-related function or changes to a component providing input to a component, which performs a safety-related function. (For this AC, a safety-related function is one that could potentially induce or allow a major, hazardous, or catastrophic failure condition to go undetected).

**(c)** Changes to characteristics of the development environment impacting the executable object code.

**(d)** Changes to memory allocation requirements so that memory margins are adversely impacted (for example, less than 5 percent margin remaining).

**(e)** Changes to timing requirements so that timing margins are adversely impacted (for example, margins are unpredictable or less than 10 percent margin remains).

**(f)** Changes to input/output requirements (such as bus loading) so that input or output performance is adversely impacted (for example, less than 5 percent margin remains).

**(g)** Data and control coupling characteristics are adversely impacted (for example, to the extent that more than 50 percent of the coverage analysis must be redone).

**(h)** Changes to interface characteristics.

**d.** Additionally, the following items should be identified in the change impact analysis:

**(1) Updates** needed to assure that the software change(s) is incorporated in the appropriate software life cycle data, including requirements, design, architecture, source and object code, and traceability.

**(2) Verification activities** needed to verify the changes and that there are no adverse effects on the system. The change impact analysis should cover how changes that could adversely affect safe operation of the system or aircraft will be verified, so the changed and unchanged software will continue to satisfy their requirements for safe operation. These verification activities may include reviews, analyses, regression testing, requirements-based testing, flight testing, and so on, including reevaluation of existing analyses, reexecution of existing tests, and new test procedures and cases (for added functionality or previously deficient testing).

**e.** When the applicant or integrator makes changes to the RSC without the RSC developer's assistance, they become responsible for the entire component.

**f.** Changes to a RSC as a result of an airworthiness directive (AD) must be coordinated with the RSC developers, users of the RSC, and the appropriate certification authority to determine how the AD applies to other projects.

## 14. CONCURRENT USE OF AN RSC.

**a.** Sometimes an RSC may be developed for use by concurrent projects. The development of the RSC and the multiple applications using the RSC may progress at the same time. In this situation, the RSC developer must create a "Reuse Plan" which includes:

**(1)** Applications and projects that will use the RSC (including the original applicant).

**(2)** The schedule for the multiple applications and projects.

**(3)** A proposed reuse approach, based on this AC's guidance and the specific project needs. The reuse plan should thoroughly address Sections 5 and 6 of this AC. The reuse approach should also propose a way to efficiently utilize FAA and DER resources.

**(4)** A list of all data items (with specific configuration identification) being developed for each user.

**(5)** A summary of which data items will be the same for all users and which data items are user-specific.

**(6)** An explanation of data items that differ among users (these may not be suitable for reuse).

**(7)** A list of affected applicants and certification offices.

**(8)** A description of how users will be enabled to use the product correctly (for example, a user's guide or interface document).

**(9)** A description of how the users will be kept up-to-date during the development and deployment of the product. For example, describe how the users and customers will be informed of problems found with the RSC, potential safety issues, and other relevant reporting processes.

**b.** The "Reuse Plan" must be coordinated by the RSC developer with all appropriate certification authorities, applicants, and integrators. All stakeholders must agree on the approach for concurrently using the RSC.

DRAFT

David W. Hempe
Manager, Aircraft Engineering Division
Aircraft Certification Service

## APPENDIX 1 – DEFINITIONS OF TERMS

For purposes of this AC, the RTCA/DO-178B Annex B definitions and the following definitions apply:

**a. Access mechanism** is the manner in which a software component is called upon to perform its intended function. This includes invocation mechanisms and data flow to and from the component. This is typically part of the interface description data.

**b. Applicant** is the manufacturer seeking certification of the product or authorization of the equipment. The applicant may be applying for a TC, STC, ATC, ASTC, or TSO authorization.

**c. Certification Authority** is the organization or person responsible within the state or country for the certification of compliance with the requirements. The certification authority is typically the FAA or foreign certification body engineer.

**d. Credit** is compliance to one or more RTCA/DO-178B objectives supported by RTCA/DO-178B software life cycle data. This compliance is used to show that the certification basis has been met and the equipment may receive a certificate. Three types of credit are referred in this AC:

**(1)** Full credit – fully meets the RTCA/DO-178B objective and requires no further activity by the user.

**(2)** Partial credit – partially meets the RTCA/DO-178B objective and requires additional activity by the user to complete compliance.

**(3)** No credit – does not meet the RTCA/DO-178B objective and must be completed by the user for compliance.

**e. Documentation configuration** is the numbering used to identify the configuration of documents used in the development process.

**f. Integrator** is the manufacturer responsible for integrating the reusable software component into the target computer and system with other software components.

**g. Installation procedures** are procedures used to install the reusable software component. These might be documented in the porting description data, interface description data, or similar data.

**h. Interface description data** identifies the interface details of the reusable software component. It is provided by the reusable software component developer to the integrator and applicant. The interface description data should explicitly define what activities are required by the integrator and/or applicant to ensure that the reusable software component will function in accordance with its approval basis.

**APPENDIX 1, DEFINITIONS OF TERMS (CONTINUED)**

**i. Maintenance code** is code residing in a airborne computer-based system that interfaces with an onboard maintenance computer or computer used by maintenance personnel. The function of this code is usually to report to the maintenance computer any problems detected during normal operations.

**j. Porting description data** is data that contains assumptions and limitations on the reuse of the component that must be followed by the user, installer, and/or integrator to ensure correct functioning of the component in a new environment.

**k. Reusable software component (RSC)** is the software, its supporting RTCA/DO-178B software life cycle data, and additional supporting documentation being considered for reuse. The component designated for reuse may be any collection of software, such as, libraries, operating systems, or specific system software functions.

**l. RSC developer** is the manufacturer of the RSC.

**m. Settable parameters** are software component data that are set before execution of the component.

**n. Software component** is some part of the airborne system's software. It is usually defined as performing specific functions within the system.

**o. Software life cycle data** is data produced during the software life cycle to plan, direct, explain, define, record, or provide evidence of activities (see RTCA/DO-178B, Section 11.0). Sections 11.1 through 11.20 of RTCA/DO-178B describe different kinds of software life cycle data.

**p. Stakeholders** are all the persons and groups involved in the development, integration, and acceptance of the RSC. Stakeholders in this AC are the RSC developer, integrator, applicant, and certification authority. One or more manufacturers may assume the roles of the RSC developer, integrator, and applicant.

**q. Target computer** is the physical processor that will execute the program while airborne.

**r. Target computer environment** is the target computer and all its support hardware and systems needed to function in its actual airborne environment.

**s. Target environment** is the same as target computer environment (above).

**t. Variables** are named memory locations that contain data that will change during software execution.

DRAFT

## APPENDIX 2 - ACRONYMS

The following acronyms are used in this AC:

| | |
|---|---|
| **AC** | Advisory Circular |
| **AD** | Airworthiness Directive |
| **ASTC** | Amended Supplemental Type Certificate |
| **ATC** | Amended Type Certificate |
| **CFR** | Code of Federal Regulations |
| **CSTA** | Chief Scientific and Technical Advisor |
| **DER** | Designated Engineering Representative |
| **FAA** | Federal Aviation Administration |
| **PSAC** | Plan For Software Aspects Of Certification |
| **RSC** | Reusable Software Component |
| **SAS** | Software Accomplishment Summary |
| **SCI** | Software Configuration Index |
| **SCMP** | Software Configuration Management Plan |
| **SDP** | Software Development Plan |
| **SQA** | Software Quality Assurance |
| **SQAP** | Software Quality Assurance Plan |
| **STC** | Supplemental Type Certificate |
| **SVP** | Software Verification Plan |
| **TC** | Type Certificate |
| **TSO** | Technical Standard Order |

## APPENDIX 3 – SAMPLE FORMAT FOR RSC DEVELOPER'S TABLE

| DO-178B Obj # | Objective Description | Credit Sought | Assumption | Means of Compliance for the Objective | Activities Remaining For Integrator/Applicant |
|---|---|---|---|---|---|
| 1-1 | Software development and integral processes activities are defined. | Note 1 | Note 2 | Note 3 | Note 4 |
| 1-2 | Transition criteria, inter-relationships and sequencing among processes are defined. | Note 1 | Note 2 | Note 3 | Note 4 |
| 1-3 | Software life cycle environment is defined. | Note 1 | Note 2 | Note 3 | Note 4 |
| 1-4 | Additional considerations are addressed. | Note 1 | Note 2 | Note 3 | Note 4 |
| 1-5 | Software development standards are defined. | Note 1 | Note 2 | Note 3 | Note 4 |
| ETC. | | | | | |

NOTE 1: Include if FULL, PARTIAL, or NO credit is being sought for the RSC. See Section 5c(1) of this AC.

NOTE 2: List all assumptions made for the credit claim. See Section 5c(2) of this AC.

NOTE 3: List data that documents the compliance to this objective. See Section 5c(3) of this AC.

NOTE 4: List the activities remaining for the integrator and/or applicant to complete the objective. This should be in enough detail that the integrator and/or applicant and the certification authority can clearly understand what remains for the overall acceptance of the system using the RSC. See Section 5c(4) of this AC.