Software Quality Assurance Plan (SQAP) in DO-178C: Ensuring Compliance and Confidence in Airborne Software

1. Introduction

In the high-stakes world of aviation software, the margin for error is virtually zero. Aircraft systems operate in dynamic, unpredictable environments, and software failures can have catastrophic consequences. To manage this risk, the **DO-178C** standard provides a framework for the development and verification of airborne software. A cornerstone of this framework is the **Software Quality Assurance Plan (SQAP)**.

The SQAP is not simply a checkbox on the path to certification—it is a systematic, structured plan that ensures that software development and verification are conducted with discipline, accountability, and consistency. It provides the organizational structure and procedures to verify that the DO-178C processes themselves are being followed.

2. Purpose and Regulatory Context

The purpose of the SQAP is to define the **quality assurance activities** necessary to ensure that the software life cycle processes comply with the plans, standards, and procedures defined for the project. It supports a number of DO-178C objectives, particularly those found in **Table A-9**, which address software assurance, compliance, process adherence, and audit readiness.

The SQAP also serves a key regulatory function. Certification authorities such as the **FAA**, **EASA**, and other civil aviation regulators require objective evidence that software has been developed in accordance with approved plans and that deviations are properly managed. The SQAP provides this evidence through documentation of **audits, reviews, corrective actions**, and **independent assessments**.

The level of detail and rigor in the SQAP is determined by the **Design Assurance Level (DAL)** of the software, ranging from **DAL A** (catastrophic failure potential) to **DAL E** (no impact on safety). For higher DALs, the independence and frequency of quality assurance activities must be increased, and the SQAP must describe how this independence is maintained.

3. Structure and Content of the SQAP

A complete SQAP should include the following major sections:

- **Introduction and Scope**
- **Applicable Documents and Standards**
- **Quality Assurance Objectives**
- **Software Lifecycle Process Overview**
- **Quality Assurance Activities**
- **Roles and Responsibilities**
- **Noncompliance Handling**

- **Reports and Records**
- **Tools and Resources**
- **Schedule and Milestones**
- **Interfaces with Other Plans**
- **Audit Procedures and Criteria**

Each section should be tailored to the project's software scope, DAL, and certification authority expectations. Importantly, the SQAP should not repeat content from the Software Development Plan (SDP), Software Verification Plan (SVP), or Software Configuration Management Plan (SCMP), but rather reference and cross-validate those plans.

4. Software Quality Assurance Processes

The core of the SQAP lies in its definition of **quality assurance processes** that operate across all stages of the software lifecycle. These include:

- **Process Audits**: Assessing whether development and verification activities are being conducted according to the SDP, SVP, and SCMP.
- **Product Audits**: Reviewing deliverables (requirements, design, code, test results) for completeness, consistency, and compliance.
- **Milestone Reviews**: Conducted at defined gates, such as Preliminary Design Review (PDR), Critical Design Review (CDR), Test Readiness Review (TRR), and Software Verification Review (SVR).
- **Corrective Action Tracking**: Identifying and resolving deviations or nonconformities.

These processes must be **proactive and ongoing**, not reactive. Quality assurance under DO-178C is not a final gatekeeper—it is a parallel activity that ensures compliance throughout development and verification.

5. Lifecycle Phase Coverage

The SQAP must outline how quality assurance will be applied during each major **lifecycle phase**, including:

- **Requirements Engineering**: Ensuring traceability, completeness, and verifiability of requirements.
- **Design**: Confirming that design reviews are performed and documented in accordance with standards.
- **Coding**: Verifying that coding standards are followed and code reviews are independently performed.
- **Verification**: Checking that test cases are requirements-based, traceable, and executed under controlled conditions.

Each phase should have **entry and exit criteria**, defined quality gates, and corresponding review activities documented in the SQAP. The plan must also state how records of these reviews are maintained and made available to stakeholders and certification authorities.

6. Roles and Responsibilities

One of the most critical elements of the SQAP is the definition of **SQA roles** and their required **independence** from development and verification. These roles include:

- **Software Quality Assurance Engineer**: Responsible for executing the SQAP, conducting audits, and generating QA reports.
- **QA Manager**: Oversees the QA program and resolves escalated noncompliances.
- **Process Owners**: Interface with QA personnel to ensure the applicability and correct implementation of defined processes.

The SQAP must explain how **organizational independence** is maintained—especially important for DAL A and B—by ensuring that QA personnel report outside the development chain of command.

7. Noncompliance Management

The SQAP must define the process for managing **noncompliances** or **process deviations**, including:

- How issues are identified (e.g., audit findings, peer reviews, test discrepancies)
- How they are logged and tracked (e.g., in an issue-tracking tool or audit log)
- The escalation and resolution path (e.g., corrective action review boards)
- The criteria for closure and re-verification

For deviations from DO-178C objectives or planned activities, the SQAP may require that **deviation requests** be submitted to certification authorities or DERs for approval. This process must be clearly documented in the plan.

8. Tools, Templates, and Reports

The SQAP should list any **tools** used to support quality assurance activities, such as:

- **Audit tracking tools** (e.g., JIRA, Helix ALM)
- **Document management systems**
- **Metrics collection dashboards**

It should also reference standard **templates** and **checklists** used in audits and reviews. For example:

- Requirements Review Checklist
- Coding Standards Compliance Checklist
- Test Case Completeness Checklist
- Process Compliance Audit Template

In addition, the plan must define how and when **quality assurance reports** are generated. These may include:

- Weekly or milestone QA status reports
- Audit reports
- Corrective action summaries
- Compliance verification logs

9. Integration with Other DO-178C Plans

The SQAP must describe how it interfaces with the other core planning documents:

- **SDP**: The SQAP ensures that the development processes in the SDP are followed and audited.
- **SVP**: It verifies that the verification strategy is implemented as described and that results are traceable.
- **SCMP**: It confirms that configuration management controls are in place, such as versioning and baseline control.

Cross-referencing these documents promotes consistency and reduces duplication, which is essential for passing audits and certification reviews.

10. Common Pitfalls and Best Practices

10.1 Pitfalls

- Treating QA as a late-stage activity
- Assigning QA to individuals who are not independent of development
- Failing to define measurable audit criteria
- Inadequate documentation of nonconformities and resolutions

10.2 Best Practices

- Involve SQA early in planning and process definition
- Maintain a consistent audit trail from process to product
- Use objective metrics (e.g., % of plans audited, # of open findings)
- Periodically review and update the SQAP based on project feedback

11. Certification Considerations

From a certification perspective, the SQAP is a critical artifact reviewed by **FAA Designated Engineering Representatives (DERs)** or **EASA software specialists**. They expect the SQAP to demonstrate:

- That SQA is **independent**, structured, and resourced
- That QA activities are **planned and executed** in alignment with DAL

- That **issues are logged, tracked, and closed** according to defined criteria
- That QA records can be **produced on demand** during audits

The SQAP also supports preparation for the **Software Accomplishment Summary (SAS)** and the **Stage of Involvement (SOI) reviews** required by certification authorities.

12. Conclusion

The **Software Quality Assurance Plan (SQAP)** is a foundational component of the DO-178C software certification framework. It ensures that all life cycle processes are conducted in compliance with established plans and standards, and that any deviations are systematically identified and resolved.

More than just a compliance requirement, the SQAP embodies the principle of engineering integrity—it helps project teams maintain discipline, improve transparency, and build confidence with certification authorities. When properly implemented, the SQAP not only helps achieve certification but also contributes to the delivery of **safe, reliable, and high-integrity software** in the aviation industry.

In the end, the value of the SQAP is not just in its content—but in the **consistent execution of the processes it defines**.

DO-178C Software Quality Assurance Plan (SQAP) Checklist

1. General Information

- Document title, ID, revision, and date
- System/project name and applicable software components
- Stated Design Assurance Level (DAL)
- Scope and purpose of the SQAP clearly defined
- References to related plans (PSAC, SDP, SVP, SCMP)

2. Referenced Documents and Standards

- DO-178C and applicable supplements (e.g., DO-330)
- Company or project-specific quality assurance standards
- Software development lifecycle standards referenced

3. Quality Assurance Objectives

- QA goals stated (e.g., compliance monitoring, process audits)
- Clear linkage to DO-178C Table A-9 objectives
- DAL-specific quality assurance activities identified

4. QA Activities Across the Lifecycle

For each phase (requirements, design, code, integration, verification):

- Entry/exit criteria defined
- QA reviews and audits described
- Frequency and scope of QA checks established
- Verification of independence between QA and development

5. Roles and Responsibilities

- Software Quality Assurance (SQA) Engineer role defined
- QA Manager or lead authority specified
- Reporting structure ensuring QA independence outlined
- Interface between QA, development, verification, and CM teams documented

6. Noncompliance Management

- Process for identifying and logging nonconformities
- Corrective action process and escalation path described
- Deviation request and approval procedures defined
- Closure and re-verification of resolved issues included

7. Audits and Reviews

- Audit types (process, product, milestone) defined
- Audit schedule aligned with lifecycle milestones (e.g., PDR, CDR, SVR)
- Audit criteria and checklists documented
- Approval gates and audit report structure explained

8. Records and Reporting

- QA review and audit reports identified as controlled artifacts
- Status reporting frequency (weekly, milestone-based) described
- Metrics collection and monitoring defined (e.g., open issues, audit coverage)
- Document retention and configuration control procedures referenced

9. QA Tools and Templates

- QA support tools listed (e.g., tracking systems, checklists, templates)
- Usage of issue-tracking tools or audit databases described
- Standard templates for audits and reviews attached or referenced

10. Interfaces with Other DO-178C Plans

- How SQAP enforces process adherence in SDP, SVP, and SCMP documented
- Coordination with Configuration Management and Verification teams outlined
- Shared roles and boundaries clarified

11. Schedule and Milestones

- Key QA milestones aligned with development lifecycle
- Defined timing for major audits and reviews
- QA visibility into all development and verification activities confirmed

12. Certification Considerations

- Demonstrates how quality assurance supports DO-178C compliance
- Prepares for SOI (Stage of Involvement) audits
- Describes availability of QA evidence for DER, FAA, or EASA review
- Cross-referenced in the Software Accomplishment Summary (SAS)

13. Continuous Improvement

- Lessons-learned feedback loop or process review strategy defined
- SQAP update and revision process described
- QA involvement in post-certification reviews included (if applicable)