FAS Topic Paper (FTP)

TITLE	REVISION	REVISION DATE
FTP1050 Agile Methodology	3	26-Feb-2024

ABSTRACT/PURPOSE:

This paper provides information regarding compliance to DO-178C/ED-12C and DO-278A/ED-109A using Agile Methodology, including clarification that Agile development frameworks like other development styles are allowed, and providing identification of potential compliance issues to be addressed if implementing the Agile principles.

RELATED DO/ED DOCUMENTS:

_**X**__ DO-178C/ED-12C: SW Airborne Sys & Equip
_**X**__ DO-278A/ED-109A:SW (CNS/ATM) Systems
_**X**__ DO-248C/ED-94C: Supporting Information
_**X**__ DO-330/ED-215: Software Tool Qualification Considerations
_**X**__ DO-331/ED-218: Model Based Development & Verification Supplement
_**X**__ DO-332/ED-217: OO Technology and Related Techniques Supplement
_**X**_ DO-333/ED-216: Formal Methods Supplement
____ Other

For internal use only—This paper is based on internal FAS FTP1050 Revision 11

Any FAS Topic Papers released by FAS have been coordinated among the members of the FAS group and have been approved by the FAS executive management committee for release.

These papers do not constitute official policy or position from RTCA / EUROCAE or any regulatory agency or authority. These documents are made available for educational and informational purposes only

FAS Team Definition and Goals:

The FAS user group monitors and exchanges information on the application of the following "software document suite" that was developed by joint RTCA/EUROCAE committee SC-205/WG-71:

- DO-178C/ED-12C - Software Considerations in Airborne Systems and Equipment Certification
- DO-278A/ED-109A - Software Integrity Assurance Considerations for Communication, Navigation, Surveillance and Air Traffic Management (CNS/ATM) Systems
- DO-248C/ED-94C - Supporting Information
- DO-330/ED-215 - Software Tool Qualification Considerations
- DO-331/ ED-218 - Model Based Development & Verification Supplement
- DO-332/ED-217 - Object Oriented Technology and Related Techniques Supplement
- DO-333/ ED-216 - Formal Methods Supplement

The goals of the FAS user group are as follows:

1. To share lessons learned in the use of the RTCA/EUROCAE "software document suite" and to encourage good practices and promote the effective use of RTCA's and EUROCAE's publications.
2. To develop FAS Topics Papers (FTPs) relative to RTCA's and EUROCAE's publications or other related aeronautical software industry topics. These FTPs may include clarification to the "software document suite" or a discussion on a new topic.
3. To identify and record any issues or errata showing the need for clarifications or the need for modifications to the "software document suite".

The FAS user group does not have the authority to change the content of any approved RTCA/EUROCAE documents. Any publications of the FAS user group may be taken into consideration by a future RTCA/EUROCAE working group.

The text contained in this document is not to be construed as guidance, but is to be used for informational or educational purposes only.

Abstract / Purpose of the FAS Topic Paper:

This paper provides information regarding compliance to DO-178C/ED-12C and DO-278A/ED-109A using Agile Methodology, including clarification that Agile development frameworks like other development styles are allowed, and providing identification of potential compliance issues to be addressed if implementing the Agile principles.

FTP Discussion:

1.0 Introduction

There are no specific life cycle methodologies specified or banned for use by DO-178C/ED-12C and DO-278A/ED-109A, however, meeting the objectives for these standards can have challenges while using Agile principles. This paper provides high-level clarifications for the challenges presented by Agile and its associated frameworks.

For safety-critical software, it is necessary to provide working software as well as compelling evidence that the software performs its intended function with an appropriate level of confidence in the software life cycle processes and their outputs. It is worth emphasizing that DO-178C/ED-12C and DO-278A/ED-109A deliberately do not prescribe any specific software life cycle, but rather that the chosen software life cycle will need to satisfy all the applicable objectives.

This paper does not provide any detail regarding how to implement Agile principles or any software life cycle for development of safety critical software.

This paper utilizes generic software engineering terms whenever possible to help those new to the DO-178C/ED-12C and/or DO-278A/ED-109A domains.

2.0 Glossary

As defined in the context of this paper:

> Aeronautical software - software that is applicable to the DO-178/ED-12 and DO-278/ED-109 guidance documents.
>
> Artifact – see “Data” definition/reference below.

As defined in DO-178C/ED-12C and DO-278A/ED-109A:

> “Baseline – The approved, recorded configuration of one or more configuration items, that thereafter serves as the basis for further development, and that is changed only through change control procedures.”

Data – *An actual definition is not provided but the following reference found in Section 11 expresses its meaning:* "Data is produced during the software life cycle to plan, direct, explain, record, or provide evidence of activities. This data enables the software lifecycle processes, system or equipment certification, and post-certification modifications of the software product."

"Independence - Separation of responsibilities which ensures the accomplishment of objective evaluation. (1) For software verification process activities, independence is achieved when the verification activity is performed by a person(s) other than the developer of the item being verified, and a tool(s) may be used to achieve equivalence to the human verification activity. (2) For the software quality assurance process, independence also includes the authority to ensure corrective action."

"Transition Criteria – The minimum conditions, as defined by the software planning process, to be satisfied to enter a process."

DO-178C/ED-12C and DO-278A/ED-109A have some differences in terminology, for example "certification" versus "approval". For the readability of this paper, only DO-178C/ED-12C terms are used.

3.0 Clarification With Respect to The Manifesto

"The Agile Manifesto" [1] and its key concepts emphasize the following:

1. Value "Individuals and interactions over processes and tools"
2. Value "Working software over comprehensive documentation"
3. Value "Customer collaboration over contract negotiation"
4. Value "Responding to change over following a plan"

This paper will further examine these concepts that can lead the safety critical regulatory development organizations to have concerns for the use of Agile. These Agile Manifesto values may in some respect be considered contradictory to some aspects of aerospace standards such as DO-178C/ED-12C and DO-278A/ED-109A.

The four key concepts are discussed below regarding potential certification concerns for satisfaction of DO-178C/ED-12C and DO-278A/ED-109A. In fact, some of these concepts identified as being de-emphasized within Agile methodology may be considered critical to the satisfaction of objectives within DO-178C/ED-12C and DO-278A/ED-109A.

1. With software development, the sequencing of the activities followed by developers is life cycle dependent. Agile promotes self-organizing teams that engage in frequent communication to implement work rather than emphasize the flow of work.

   Within DO-178C/ED-12C and DO-278A/ED-109A, the gating of this flow is captured as the transition criteria which defines the sequences of the activities that are used in the life

cycle. The transition criteria concept is mandatory at higher assurance levels but allows flexibility in defining the gates to step through the life cycle workflow, which permits Agile and other life cycle models.

2. Agile methodologies generally do not prescribe any specific documentation as seen in the second value of the Agile Manifesto which focuses on working software over complete documentation.

   DO-178C/ED-12C and DO-278A/ED-109A standards require artifacts that are a consequence of the process as evidence of satisfaction of the objectives. Although artifacts are required to be identified as evidence, there are no strict packaging requirements for them. This allows for the organization and packaging of artifacts that align with the chosen development methodology. Although the time in which artifacts are completed and consolidated may be delayed, they must be performed for each baseline to enable the demonstration of objective compliance for a given software baseline.

3. Agile promotes this collaboration exchange, however Agile may be viewed as de-emphasizing the captured documentation of the agreement. Interactions with stakeholders are critical to ensuring the software development captures the appropriate system intent. There are no objectives that address contractual aspects between stakeholders in DO-178C/ED-12C and DO-278A/ED-109A, and as such this Agile value is not in contradiction with DO-178C/ED-12C and DO-278A/ED-109A. However, DO-178C/ED-12C and DO-278A/ED-109A require artifact-based evidence; and although the interaction and exchange of information with stakeholders is essential, one must also capture that technical information for evidence of the agreement (e.g., specification, derived requirement assessment, and problem reports).

4. Emphasis on development teams responding to requirements, process, and tool changes with the mindset of customer priorities versus strictly executing to the plans is the fourth high-level value of Agile. Having teams working efficiently while improving the process being executed is a positive goal.

   However, process changes may introduce the risk of inadvertently affecting compliance with DO-178C/ED-12C and DO-278A/ED-109A if the changes affect artifacts used for certification credit. The process or tool changes should be documented to reflect the resultant process execution workflow and communicated to the teams, and require an assessment of the impact on the existing life cycle data and compliance to DO-178C/ED-12C and DO-278A/ED-109A. Depending on the magnitude of changes and risk level of noncompliance due to the changes, the Plan for Software Aspects of Certification (PSAC) and possibly the corresponding plans and standards may need to be updated and reapproved; otherwise changes need to be captured and justified in the Software Accomplishment Summary (SAS). As such, the process to address changes to the plans and standards should be documented in the plans; thus, gaining agreement with the certification authority on their method of visibility to changes.

4.0 General Clarifications

Due to the different frameworks of Agile it is not possible to cover all potential clarifications. General clarifications include but are not limited to:

- When working in an Agile framework, there are typically iterative builds of the software. This development process may be treated as informal with incremental growth of requirements through implementation and trial executions. As the understanding of the behavior grows, so should the rigor of the definition of the artifacts. The requirements and other artifacts will reach a level of maturity that will justify the additional expenditure of resources to capture the evidence formally. Determining the appropriate time for this transition into formal activities needs careful consideration. The informal nature of a process and its implications should be well understood to assure the informal activities, which will not be used for certification credit, do not prevent or hinder compliance demonstration in the formal activities. At this point it may be decided that some of the requirements may be deferred for future product releases. Although this continuous definition of requirements is no different in other life cycles used within DO-178C/ED-12C and DO-278A/ED-109A, baseline(s) will need to be established if any certification credit is sought for verification activities related to these builds. Such verification activities requiring baselined development data include but are not limited to: requirement reviews, design reviews, code reviews, and testing. Furthermore, bi-directional traceability through the software development artifacts needs to be established and maintained, which originate from and include the system level requirements.

- Any changes to baselined life cycle data, including data coming from the customer (e.g., system requirements) will require an assessment of impact per the change control process. As such, changes could affect work done in previous iterations. Due to the focus on rapid changes within Agile iteration cycles, particular attention should be placed on change control and adherence to the transition criteria in the approved project plans.

- Many software development methodologies use the term baseline to mean a snapshot of all artifacts at a particular point in time. However, DO-178C/ED-12C and DO-278A/ED-109A use the term for the approved, recorded configuration of one or more configuration items, that serves as the basis for further development, and that is changed only through change control procedures, see DO-178C/ED-12C and DO-278A/ED-109A Paragraph 7.2.2. Therefore, once a baseline is established for an artifact, it is necessary to formally maintain issues through DO-178C/ED-12C and DO-278A/ED-109A problem reporting, change control, and change review processes.

- Rigorous problem reporting and documentation of whether specific problems were addressed, or not addressed, may not be present in Agile developments. For aeronautical software, bookkeeping of problem reports and their management are necessary after artifacts are baselined to: a) evaluate that the planned processes have been followed, and b) evaluate the final product to determine the effect of unresolved problems on the operation of the product. As such, when verification activities are performed for

certification credit, the required rigor of problem reporting, change control and change review on the baseline artifact(s) commences.

- Embedded QA personnel are often used in Agile development, and their role is typically different than the QA role described in DO-178C/ED-12C and DO-278A/ED-109A.

  The purpose of DO-178C/ED-12C and DO-278A/ED-109A QA is to ensure that the documented process plans and standards are followed. This ensures that the activities used to satisfy the objectives are accomplished.

  An Agile process that embeds QA functions as intrinsic activities will need to establish a documented level of independence to ensure appropriate separation between development/verification and assurance.

- In Agile development frameworks, QA may overlap with testing and other development activities. Independence in DO-178C/ED-12C and DO-278A/ED-109A (see Glossary for definition) requires QA to have separate responsibilities from the development and testing teams. In addition, testing, review, and analysis activities should be independent from the personnel who authored the artifact that is being verified.

- The SAS should include all information that is needed to confirm that the plans as described in the PSAC have been followed, and adequately describe how the objectives have been met. Any deviations should be addressed and justifications for such deviations provided.

  - While having iterative development is not unique to Agile, this is the central characteristic of Agile that needs attention in terms of compliance. As the processes and requirements could be modified for different iterations, the SAS would then need to include a summary of development/verification to adequately describe the changes that have occurred throughout the life cycle for which certification credit is sought.

  - Agile frequently uses a backlog to track work to be accomplished, including functions to be implemented, corrected, removed, etc. Care should be taken to ensure that backlog items that qualify as DO-178C/ED-12C and DO-278A/ED-109A problem reports are managed and controlled as such. These problem reports should be confirmed to include deficiencies from the requirements, their implementation, and their verification. All open/unresolved problem reports need to be included in the SAS, and as noted above, need to include justification to remain open.

5.0 Summary

Agile concepts promote flexible approaches and place less value on comprehensive documentation during the development stages. Agile and its associated frameworks do not inhibit the satisfaction of artifact creation but do de-emphasize them.

Safety critical concepts promote clear plans, artifact-based evidence, and deterministic systems. DO-178C/ED-12C and DO-278A/ED-109A require plans and artifacts as evidence for completion of objectives that are used to obtain certification credit. Nevertheless, these standards do allow separating the interim artifacts, such as preliminary informal testing, from the formal testing performed at a later stage.

Agile and other development life cycle methodologies can be adopted and utilized to accomplish the necessary level of confidence needed in the safety-critical world.

However, the level of rigor should leave no doubt that the documented software verification process was followed and any changes or deviations from the planned processes were benign. The documentation should provide the confidence necessary that all life cycle data artifacts are complete and correct.

6.0 References

[1] Agile Manifesto, http://agilemanifesto.org (accessed January 2020)