Airborne Software Change Control Process and Strategies of Civil Aircraft

Chen Zhao¹*, Yi Chu²

¹Shanghai Aircraft Design and Research Institute, Shanghai, 021-20866386, China
²Shanghai Aircraft Design and Research Institute, Shanghai, 021-20866384, China
*Corresponding author’s e-mail: zhaochen@comac.cc

Abstract. In the modern civil aircraft, the number and complexity of airborne software are increasing rapidly. The software may change at any stage of the development process and the change process should be under control. This article analyzes the basic unit of software change control and the configuration identification used for airborne software, then introduces a process and strategies of software change control for civil aircraft.

1. Introduction
Airborne software is the software that is installed in aircraft equipment and is part of the aircraft design. Airborne software controls all the key systems from flight control, radar to engine, which is one of the core technologies of aircraft. As an intangible product, it has high complex content, frequent version update and huge related engineering data. Especially for civil aircraft, airworthiness requirements are very strict. Faced with dozens or even hundreds of software in the same aircraft model, establishing a unified scientific software change control process is indispensable and important in the process of aircraft development and certification.

This article intends to discuss and introduce the control and management of the Airborne software modification process from the perspective of the main manufacturer.

2. Basic unit of change control
The premise of software change control is to determine the basic control unit of software. Software is distinguished and identified according to the basic requirements and functional units of software. Configuration items are used as the basic unit of change control.

2.1. Configuration item
Airborne software configuration item can be defined as software unit that can independently perform configuration management in the process of product structure decomposition.

The change control of the airborne software is to manage the designed software modules in the software life cycle with configuration items as the unit.
Figure 1. Schematic diagram of two levels of configuration items

The configuration item of software consists of the following two types as figure 1:

- Line replaceable unit/module containing software.
- Software life cycle data.

Each specific software should contain these two parts of information to identify the configuration. Hardware identification usually includes hardware part number, and software identification usually includes software part number and/or software version number. Software life cycle data includes plan, development, verification, quality assurance and configuration data, as well as related records and tracking items which identified by data no and data version.

2.2. Organization form of software configuration items

In the process of software change control, due to the particularity of airborne software, Software can not exist independently from the specific hardware equipment, so the configuration items of software generally can not be separated from the specific product structure tree, and usually exist in the form of sub nodes in the product structure.

For a certain software configuration items, there is always a part of data that remains unchanged in the whole life cycle of the product, which is the basic information data; and there is another part, every time the software changes, this part of data may change accordingly, which is traced and identified by the configuration item identification. Software configuration item instances are used to manage the second part. The figure 2 shows a typical software product management mode. The top-down level of the product structure is mainly divided into system level, equipment layer and software level. The software configuration items node contains the basic information data and several configuration items instance. Configuration items instance marked with configuration identification.
Two management methods can be used for configuration identification:

- When the lower level identification changes, all the upper level identification will be automatically updated. Only the identification of the layer itself needs to be checked to distinguish the configuration status of special layer.
- If the lower level identification changes, it does not update any upper level identification. It must traverse all the lower layers to check whether all the identification are consistent to distinguishes the configuration state of special layer.

3. Change control process
Change control process refers to the specification and process that must be followed in the whole process from the delivery of software to the final configuration. The figure 3 is a compact software change control flow diagram introduced for civil aircraft.
3.1. **Software delivery**

The supplier shall ensure that the airborne software development process, including changes before and after software delivery, is under its quality assurance and configuration management control. During software delivery, the supplier shall submit data that can fully describe the configuration of the delivered software, including but not limited to the following forms:

- Software change record sheet.
- Software configuration description document.

The design department shall sign and approve the configuration description documents.

Software delivery is divided into two categories: delivery with equipment and field loading. After software delivery, the supplier's changes to the delivered software will be controlled by the configuration of the main manufacturer.

3.2. **Software change review**

After the software is changed, the following aspects need to be analyzed at least:

- Whether software changes affect the security of the system.
Whether the software is under sufficient configuration control of the supplier to ensure that the software before change can be recovered when necessary, and that all changed parts are clearly defined.

Will any tests need to be repeated?

Whether the software changes affect the conclusion of the completed test of the system and related systems, resulting in invalid previous test conclusion, and whether it is necessary to supplement the corresponding laboratory test and on-board ground test. If the test results of relevant systems are affected, the design department shall actively coordinate with relevant systems and assist them to formulate solutions.

Will the airplane maintenance manual need changes?

Will the aircraft flight manual need changes?

Compatibility impact based on software compatibility matrix of the aircraft.

3.3. Configuration accounting
Configuration accounting is a real time record of the status of each software change. Configuration accounting occurs in each software change and needs to maintain continuity and traceability.

After the software delivery, the design department must fill in the software configuration accounting information according to the documents submitted by the supplier at the time of delivery. A typical configuration accounting file is the configuration accounting table, which contains the configuration items and configuration identification information of software, and usually includes the location of software or equipment, software development assurance level, restricted use status, applicable sorties and other information.

3.4. Software field loading
Software field loading means that the supplier can deliver the executable object code of the software separately, and load the software into the target device through specific equipment, connection mode and loading process. Not all software supports field loading. In modern highly integrated civil aircraft projects, about 1/2 to 1/3 of the software can be field loaded. For this part of the software, in addition to the software configuration accounting, it also needs to follow step by step specific loading process, and produce the relevant record file. A typical software loading record file is software change record table, which is invented and designed as two parts: software loading application and software loading confirmation in engineering practice.

3.4.1 Software loading application
Software loading application shall be submitted in advance. The applicant shall fill in the application part of the software change record table and submit the records related to the software change and the configuration description document of the changed software. The applicant shall take the record and documents as the attachment of the software change record table. After the application part is signed and proved, the data shall be archived, and the airworthiness department shall coordinate with the certification authority whether to witness the loading process.

3.4.2 Software loading confirmation
When the applicant loads the software, Qualified responsible person must be appointed to witness on site. The executor and witness of the software loading shall sign in the software change record to confirm the software loading process and result, and take the loading record as the attachment of the software change record table. After the software loading is completed, the configuration control board shall complete the signature of the loading confirmation part of the software change record table. The software configuration accounting shall be performed according to the information after the software loading.
4. Conclusion
The change control of airborne software runs through the whole process of civil aircraft development, lab test, flight test and certification. It is an important part of airborne software quality assurance and the basis of airborne software compliance verification. In the project practice of a civil aircraft, through determining and establishing the software products and configuration items of the whole aircraft, establishing and improving the process and strategies of software change control, and applying it to the actual management of airborne software, This process and strategies of airborne software change control proved effective, which ensures the progress of the whole development of airborne software.

References
[1] Tong, Y.W., Sun, J.H. (2017) Research on airborne software change strategies for civil aircraft manufacturer. Civil Aircraft Design & Research, 2017, 2: 94-99.
[2] Ji, J., Zhang, D.H., Li, S. (2011) Configuration management and change control method for civil aircraft engine. Compute Integrated Manufacturing Systems, 17, 11: 2514-2525.
[3] Jiang, S.L. (2004) Engineering change control. Civil Aircraft Design & Research, 2004, 1: 20-22, 27-30.
[4] Radio Technical Commission for Aeronautics (RTCA). (2011) Software Considerations in Airborne Systems and Equipment Certification (DO-178C). RTCA Inc, Washington, D.C.