ISS Columbus Module On-Board Software Maintenance

Jasminka Matevska

Astrium Space Transportation GmbH
Airbusallee 1
28199 Bremen
jasminka.matevska@astrium.eads.net

Abstract: Ensuring correct and consistent functionality during utilization and enhancement of long-living complex space systems as the ISS Columbus Module is one of the main challenges for its maintenance approach. The Columbus Laboratory consists of many subsystems in order to assure vital and working conditions for the on-board crew and execution of different science experiments in space. The on-board systems are functioning autonomous and are supported by software systems and control centers on ground. The utilization of the systems requires processing of a huge amount of monitoring and control data in near real time. Due to criticality of some system functions, their availability has to be sustained even during performing maintenance activities coping with different components, target platforms and responsibilities. This paper provides an overview of the ISS Columbus Module On-board Software Maintenance Process Model.

1 Introduction

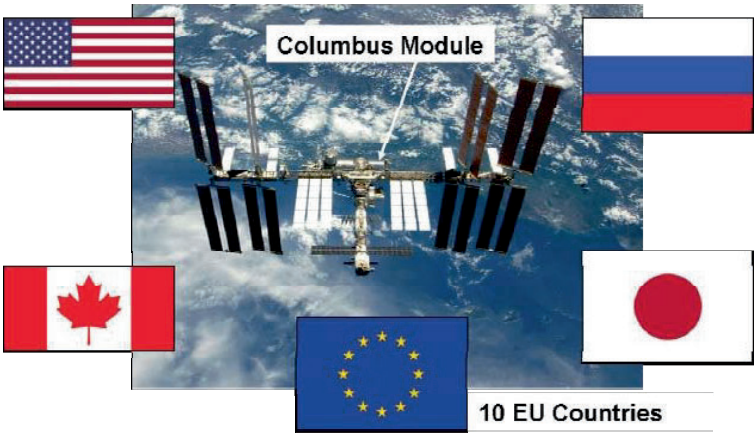

Figure 1: International Space Station (ISS)

The International Space Station (ISS) is the largest space project in history so far. It consists of different attached modules used as laboratories for science experiments in the microgravity in space. Each module provides its own system consisting of many at principle autonomous subsystems in order to assure vital and working conditions for the on-board crew. This requires processing of a huge amount of monitoring and control data in near real time.

Many countries are contributing in this project (e.g. USA, Russia, Canada, Japan, Germany, France, Italy, UK, and Spain). The European contribution to the ISS is the Columbus Module. It has been launched and attached to the International Space Station in February 2008 and shall be utilized until at least 2020. Three crew members can work and perform various science experiments on-board Columbus. In principle, the system is functioning autonomous and operated in orbit. It is commanded mainly from ground by corresponding control centers and supported by ground engineering, integration, verification and test facilities. Although Columbus is a European Module, there are many interfaces both on system and organizational level with the international partners. Monitoring, operation and commanding of the different subsystems (e.g. water pump, electrical system, data management system) is supported by communicating software systems on-board and on ground. Therefore, the Columbus Module can be considered a complex long-living space system both from the technical as well as from the organizational point of view.

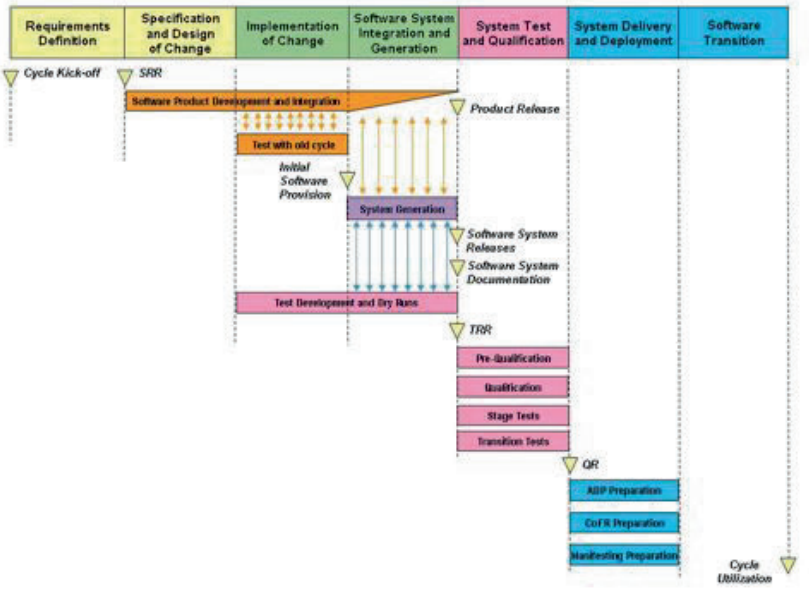

Figure 2: Columbus Module Software Cycle Approach

Due to the complexity of the system and its interfaces, it is a considerably big challenge to ensure a well functioning and consistent on-board system and supporting ground facilities over a long period of time.

During the utilization phase of Columbus, there is a necessity of efficient maintenance of all Columbus subsystems and thus maintenance of the on-board software system.

Maintenance of the Columbus on-board software system includes all types of maintenance: corrective, adaptive, perfective, and preventive. It follows the software cycle approach (Figure 2). Each cycle can be understood as a software development iteration embedded within the V-Engineering Process Model [PRD]. Due to high availability and reliability requirements of some subsystems, the preparation of a new cycle has to be performed in parallel to sustaining activities of the flight following cycle (the one installed and utilized at present). In principle, a cycle is following the phases and milestones of software development as defined in the ECSS standards (European Cooperation for Space Standardization). Included steps are described as follows.

2 Requirements Definition

All requirements of the utilized on-board software system are documented within a maintenance specification document [MSP]. All problems and enhancement requests identified during utilization are tracked via a dedicated problem tracking tool [SPR] and analyzed by corresponding boards and responsible engineers. At the beginning of one software cycle the baseline definition of the outstanding cycle is agreed and the delta of problems and enhancement requests to be implemented is identified (Baseline Kick-Off Review). According to the analysis results, additional requirements are defined or existing requirements are revised (System Requirements Review). Additionally, the verification method of all requirements is determined [MSP, VCD].

3 Specification and Design of Changes

All identified changes are analyzed and the impact to the system and corresponding system documentation is defined. All specification, design or interface changes are documented in the appropriate architecture definition and interface specification documents [ADD, ICD].

If any software interfaces to the international partners (e.g. NASA) are impacted, the impacted ISS documents are updated [ISS] and the corresponding NASA/ESA multilateral boards have to approve the changes.

Since the system is already utilized and only design changes are specified, this phase is considered completed by a Delta Design Review.

4 Implementation of Changes

After successful definition of requested changes, the detailed design of changes is defined as a foundation for the corresponding implementation work.

All impacted products are updated or extended by the responsible engineer and provided to the responsible board for release.

5 Integration and Generation of Software System

All released products are integrated in a software system assembly by a dedicated in-house developed generation environment [GEN, IWB]. The generation approach has to cope with a heterogeneous system composed of multiple software components which differ in terms of responsible organization, target platforms, and development environments. The generation environment provides all necessary assemblies of the on-board software system and all variants for ground facilities.

Nevertheless, the automatic generation of the software, release tracking, configuration management of versions and variants is not the focus of this paper, since not in our responsibility.

6 System Test and Qualification

Qualification of Columbus on-board software includes different verification methods: I (inspection), ROD (review of design), T (test) or A (analysis) [MSP, VCD].

Closeout of all requirements having "T" as verification method assigned is provided by tests on different test levels.

Product level tests are performed as part of the product change implementation.

For system level tests, multiple test facilities are used:

- Software test facility including a software simulation of electrical systems
- Flight following test facility including flight hardware
- Training, qualification and verification facility
- ISS system verification facility

As preparation for tests, all necessary test procedures are created or updated according to the change definition [TED].

Before starting with the official qualification phase, the Dry-Run test phase is performed. This phase is used to finalize the system configuration and test procedures. Identified

problems can be fixed by performing internal iterations consisting of steps as described in the sections 4, 5 and 6 (Implementation, Integration and Test).

The mandatory milestone in order to start the formal qualification phase is the Test Readiness Review. At this time point, the complete software system including test procedures and facilities shall be integrated and released. It is essential to assure a well defined configuration both of the system under test and all test facilities.

The first phase of the software system qualification campaign is called Pre-Qualification and is performed on the software test facility including a software simulation of electrical systems. The Columbus software system team aims to perform as many tests as possible on this facility in order to discover problems and provide fixes as early as possible. Problems discovered during this phase can be fixed by patching the system assemblies.

After successful pre-qualification the succeeding tests are performed on the flight following facility (Qualification) and the ISS system verification facility (Stage Tests).

For dedicated qualification of operational products and training of astronauts, the training facility is used.

The Columbus on-board software system is considered qualified for the defined cycle after successful closure of all requirements. The final milestone is the Qualification Review.

7 System Delivery and Deployment

The qualified Columbus on-board system is released and delivered for on-board deployment following the complete process of final integration, generation and release of the software.

For on-board delivery, the necessary preparation for flight is performed. All items have to provide a Certificate of Flight Readiness (CoFR) including a variety of material test reports and safety assessments (Acceptance Data Package (ADP)). For final approval for flight, all items are packed and documented within appropriate manifesting documents. After approval by the Cargo Review for the assigned spacecraft vehicle (e.g. Progress, Soyuz), they can be deployed to orbit.

8 Software Transition

On-board, the data carrier of the on-board system software is used for installation of the software. The installation of the qualified software is performed as a stepwise software transition from one software cycle to another in order to assure the availability of critical system functions. Due to technical complexity of the system and organizational complexity of supporting engineering and control centers, additional operational

preparations are done. All operational aspects are defined, documented and released in appropriate procedures [OPS]. A dedicated transition procedure is defined and transition tests are performed. Finally, the Transition Readiness Review approves the system readiness for transition.

After successful transition the utilization phase for the new software cycle can start.

9 Summary

The ISS Columbus Module is successfully utilized since its launch and attachment to the ISS in February 2008. This success is a result of efficient and high quality work of various teams on-board ISS and on ground in the supporting engineering and control centers. The cycle approach for maintenance of the Columbus on-board software system has been followed with success for many years starting with the activation cycle 10.1 as the first utilization software system cycle. At present, the cycle 13 is utilized on-board. The qualification phase of cycle 14 is the ongoing activity on ground.

References

[ADD12] Columbus On-board Software Architectural Design Documentation, Astrium Space Transportation Internal Documentation, 1998-2012

[ICD12] Columbus On-board Software Interface Control Documentation, Astrium Space Transportation Internal Documentation, 1998-2012

[PRD12] Columbus On-board Software Maintenance Process Documentation, Astrium Space Transportation Internal Documentation, 1998-2012

[MSP13] Columbus On-board Software Maintenance Specification, Astrium Space Transportation Internal Documentation, 2013

[VCD13] Columbus On-board Software Verification Control Document, Astrium Space Transportation Internal Documentation, 1998-2013

[TED13] Columbus On-board Software Test Documentation, Astrium Space Transportation Internal Documentation, 1998-2013

[SPR13] Columbus System Problem Report Database Documentation, Astrium Space Transportation Internal Documentation, 1998-2013

[OPS13] Columbus Operational Documentation, ESA/DLR/Astrium Space Transportation Internal Documentation, 1998-2013

[ISS12] NASA International Space Station Documentation, NASA/ESA Internal Documentation, 1998-2012

[GEN13] Columbus Software Generation Documentation, ESA/DLR/Astrium Space Transportation Internal Documentation, 1998-2013

[IWB10] Ignatova, T.; Westerholt, U.; Brandt, M.: Advanced Software Maintenance Approach for the Complex Columbus Flight Software System. Proc. of Data systems in Aerospace (DASIA) 2010, Budapest, Hungary, 2010; P. 289-292.

[ECSS12] ECSS Standards, European Cooperation for Space Standardization, 1993-2012