THE COMPONENT LAYER OF CORDET ON-BOARD SOFTWARE ARCHITECTURE

(1) Ana-Isabel Rodríguez, Francisco Ferrero, Elena Alaña,  
(2) Andreas Jung,  
(3) Marco Panunzio, Tullio Vardanega  
(4) Adrian Grenham

(1) GMV Aerospace and Defence (GMV), Spain, Email: airodriguez@gmv.com  
(2) European Space Agency (ESA), ESTEC Noordwijk - The Netherlands, Email: Andreas.Jung@esa.int  
(3) University of Padua (UNIPD), Italy, Email: panunzio@math.unipd.it, tullio vardanega@math.unipd.it  
(4) SciSys, U.K., Email: Adrian.Grenham@scisys.co.uk

KEYWORDS

On-board Software, Reference Architecture, Component-Based Engineering, Model Driven Architecture

ABSTRACT

This paper presents the current proposal for the CORDeT-2 On-board Software (OBSW) Reference Architecture (OBSW-RA), and outlines the design principles and process which contribute to its design. In particular, the paper addresses the OBSW-RA Component Layer of the architecture, which is a concrete instantiation of the CORDeT-2 Component Model and the tool-set that supports it. CORDeT-2 is a continuation of the CORDeT and DOMENG studies [2]. The CORDeT-2 consortium has participated in those studies and is also involved in the SAVOIR-FAIRE working group.

The SAVOIR (Space Avionics Open Interface Architecture) group [3] is an initiative between the European Space Agency, the National Space Agencies of France and Germany, and Space industry at prime and supplier level that aims to foster and integrate the stakeholders’ vision of an avionics development of the future, which maximises reuse and standardisation. The SAVOIR Advisory Group (SAG) has been established to steer the work plan of the technical discussions to be held among stakeholders. SAVOIR-FAIRE (Fair Architecture and Interface Reference Elaboration) is a specific subgroup of the SAG which holds detailed technical discussions on specific On-Board Software architectural topics, offers recommendations to Research and Development (R&D) activities and provides the SAVOIR Advisory Group with a synthesis of their results. The CORDeT-2 study is part of the ESA TRP (Technology Research Program) studies on component-based engineering.

The scope of the CORDeT study projects is the software executed on-board of the spacecraft, i.e. the On-board SW (OBSW), also referred as Flight Software, in ESA’s space missions. Software executed on-board of the spacecraft is typically hardware constrained, subject to high dependability requirements, optimised, subject to rigorous development and spacecraft and mission dependent. The software for on-board space systems [7] has to meet stringent constraints in terms of timeliness, dependability and safety.

The OBSW has seen a constant growth in size and complexity since its inception. According to NASA [8], the size of flight software is growing by a factor of ten every ten years. The current estimates for the Orion’s primary flight software exceed one million lines of code. The growth trend is expected to continue on account of increasingly ambitious requirements and because of the advantages of situating new functionality in software or firmware rather than in the hardware. In designing computer systems for space applications [7], we want to optimise the availability, capability, flexibility, and reliability of the system while minimizing cost and risk. Since space systems rely more and more on software to achieve a large amount of complex functions, cost and risk are consequently closely dependent on software.

The CORDet-2 on-board software reference architecture prescribes the principles guiding the design of the architecture, that follows the Model Driven Architecture (MDA) paradigm [6], the principles of
Component Based Software Development (CBSD), and the SAVOIR-FAIRE architectural concepts [3], recognized as the best solution to the construction of a Spacecraft software systems.

CBSD [4] allows systems to be designed through the composition of software components, whether pre-existing or yet to be developed. In order to exploit the ability of CBSD to produce correct and dependable systems with predictable quality attributes, formal component models are constructed including functional and non-functional properties that can be statically analysed to assess the quality and the performance attributes of the system model. MDA [6] is a specific initiative in the general context of the Model Driven Engineering (MDE) [5] that uses models to abstract away from implementation details and technology choices, to facilitate the port of the invariant elements of the system model to variable platforms.

As shown in Figure 1, the On-Board Software Reference Architecture (OBSW-RA) defined by COrDet-2 is split into three major parts or layers: Component Layer, Interaction Layer and Execution Platform.

- The Component Layer is implemented from the Component Model and contains instances of component implementations.
- The Execution Platform (EP) contains all the services needed to support the Component Layer, including services for components, containers and connectors. The EP contains domain-specific services, such as Monitoring and Control, as well as domain-neutral services such as message transfer services and packet services. Monitoring and Control is performed entirely in the EP and the Packet Utilization Standard (PUS) [10] represents just one possible implementation of M&C.

Note: The internal interfaces in the Execution Platform are not addressed in COrDeT-2. The Execution Platform does not assume any specific computational model: this is determined when the target platform is selected and the Execution Platform is instantiated on it.

- The Interaction Layer binds the Component Layer to the Execution Platform. The Interaction Layer is automatically generated from the system model upon the selection of the target platform and the associated technology. This ensures the portability of the elements of the Component Layer across multiple Execution Platforms.

The three-layered OBSW-RA is intended to support the implementation of general satellite and exploration spacecraft on-board software application in the Space Domain. Hence any functional feature (Space domain specific) can be reflected in the OBSW. The OBSW-RA also addresses some domain neutral aspects which are common to any embedded software domain, for example the Tasking service which is used by the Interaction Layer (component containers) to permit multi-tasking.

Components in the COOrDeT-2 vision are pure functional units; hence they can only contain functional code that specifies the sequential behaviour of the component. A component is the unit of composition in the OBSW-RA approach. The whole software is built as an assembly of components, deployed on an execution platform which takes care of their correct execution.
The Component Layer which implements the Component Model uses three distinct software entities: the component, the container and the connector. The component is the only entity that appears at design level, e.g., in the user space. Containers and connectors instead pertain to the implementation level.

Figure 2 depicts a connector that regulates the interaction between two containers, which in turn implements the functional binding between two components. The figure also shows that there can never be direct connection between a component and the underlying execution platform: the connection is always mediated by the container.

![Figure 2: A connector that realizes the communication between two containers](image)

All non-functional concerns are not addressed by the component specification and implementation; instead they are realized by the container and the connector that encapsulate the component, or, via them, by the execution platform.

A prototype implementation of the Space Component Model (SCM) meta-model and the associated editor is under development and includes the.ecore (EMF core meta-model [11]) diagrams used for the specification of the SCM meta-model.

The COrDeT-2 Development Framework chain is being built up and made available for the case study implementation. COrDeT-2 promotes well-spread and well-known technology for the implementation of the tooling, and in particular the component-oriented modelling environment:

- Model-to-Text transformation [12] and graphical modeller technology.

The COrDeT tooling provides the following features:

- Space Component Model (SCM): the SCM meta-model and the associated model editor.
- SCM Obeo Designer Editor [14]: the SCM model editor, which includes features to manage libraries of components, and the installation of multiple analysis and transformation engines.
- SCM to TASTE generator: the SCM to TASTE [9] transformation/analysis engine, which produces the necessary TASTE inputs and generates the Ada source code of some of the execution platform services.

ACKNOWLEDGEMENTS

This work was supported by the COrDeT-2 project [1] (“Interface Standardization and Verification - COrDeT-II”), ESA Technological Research Program (TRP) study, and ESTEC Contract No. 000100991/10/NL/JK. The partners involved in this study are GMV as prime contractor, SciSys, the University of Padua, SoftWcare, Thales Alenia Space and Astrium Satellites.

REFERENCES

Abstract - January 2012


[10] Telemetry and Telecommand packet utilization standard (PUS), ECSS-E-70-41A ECSS-E-70-41A


