OCORA

Open CCS On-board Reference Architecture

High Level Tooling
Gamma Release

This OCORA work is licensed under the dual licensing Terms EUPL 1.2 (Commission Implementing Decision (EU) 2017/863 of 18 May 2017) and the terms and condition of the Attributions- ShareAlike 3.0 Unported license or its national version (in particular CC-BY -SA 3.0 DE).

Document ID: OCORA-30-007-Gamma

Version: 1.10

Date: 04.12.2020

Status: Final

Revision history

Version	Change Description	Name (Initials)	Date of change
1.00	1st Gamma version based on Beta incl. comments	RM	2020-11-16
1.10	final for Gamma Release	RM	2020-12-04

Table of contents

Table of tables

References

The following references are used in this document:

[1] OCORA-10-001-Gamma – Release Notes
[2] OCORA-30-001-Gamma – Introduction to OCORA
[3] OCORA-90-002-Gamma – Glossary
[4] OCORA Memorandum of Understanding, dated March 25th, 2019
[5] OCORA Code of Conduct, dated October 20th, 2019

1 Introduction

1.1 Document context and purpose

This document is published as part of the OCORA Gamma release, together with the documents listed in the release notes [1]. It is recommended to read the OCORA Introduction [2] first and be aware of the OCORA Glossary [3].

The purpose of this document is to provide the OCORA participants with an overview of the tooling. It is and will remain a high level document with the aim to ensure common approach and basis and also give the freedom of choice in all not defined areas to the workstream leaders to define appropriate processes, methodology and tools if required for the specific type of work foreseen.

In the next phase OCORA tooling will be revisited in order to proof sector compatibility with other ongoing sector initiative (e.g. RCA, S2R-2).

The basis of this document is the OCORA contractual framework including the MoU [4] and the CoC [5].

2 Requirements for Tooling

Tool comparison classifies requirements as follows:

- M: mandatory
- 3: highest priority
- 2: strong recommendation
- 1: useful

2.1 Requirements on Tools

The following requirements are used to assess the toolchain:

ID	Requirement	Ranking
1	Formal design	M
2	Formal verification	M
3	Simulation (of design)	M
4	Animation & Visualization (Graphical User Interface)	2
5	Code generator	M
6	Code generator optimized	2
7	Traceability	M
8	Proven in use	M
9	Wide use in the industry and/or other domains	3
10	Used in multiple domains	M
11	CENELEC 50128 or IEC 61508 certification	M
12	Runnable on common PC	M
13	User-friendly	3
14	Safety Analysis[1]	M
15	Long Term Support	3

Table 1 - Requirements on tools

[1] Safety Analysis is to be interpreted as the tool providing dedicated functionality for safety certification activities of the design created in the tool.

2.2 Requirements on Design Artefacts

Tool comparison uses the requirements on design artefacts created with tools (including also some additional aspects) provided in table below.

ID	Requirement	Ranking
1	Modular	M
2	Open Interfaces	M
3	CENELEC 50128 or IEC 61508 certification	M
4	Scalable and demonstrable on a common PC	3
5	Ready for implementation on a target hardware	M
6	Functionally correct	M

Table 2 - Requirements on design artefacts

3 Comparison of Model-based Software Development Tools

This section provides evaluation and comparison of tools using different requirements:

- Requirements on tools
- Requirements on design artefacts
- Requirements on support for the V-cycle phases

A value between 1 to 9 is used, where 1 is the lowest possible value (meaning poor) and 9 is the highest (meaning excellent).

3.1 Requirements on Tools

The table below compares the tools using the requirements section.

ID	Requirement	SCADE	B Method	SPARK
1	Formal design	9	9	9
2	Formal verification & proof	7	9	9
3	Simulation (of design)	8	1	9
4	Animation & Visualization	9	6	5
5	Code generator	9	9	9
6	Code generator optimized	9	1	9
7	Traceability	7	7	N/A
8	Proven in use	9	7	9
9	Wide use in the industry and/or other domains	6/7	5	7
10	Used in multiple domains	9	3	9
11	CENELEC 50128 or IEC 61508 certification	9	9	9
12	Runnable on common PC/COTS	9	9	9
13	User-friendly	8	6	7
14	Safety Analysis	8	?	N/A
15	Long Term Support	8	7	9

Table 3 - Comparison of requirements on tools

3.2 Requirements on Design Artefacts

Comparing tools using the requirements from Table 2.

Number	Requirement	Scade	B/Event B	SPARK
1	Modular	9	9	9
2	Open Interfaces	9	9	9
3	Cenelec 50128 certification	9	9	9
4	Scalable and demonstrable on a common PC	9	9	9
5	Ready for implementation on a target hardware	9	9	9
6	Functional correct	9	9	9

Table 4 - Comparison of requirements on design artefacts

3.2.1 Requirements on Support for V-cycle Phases

It is a goal to use tool support to cover all V-Cycle phases and to enable certification of systems developed, according to requirements on SIL 4 in CENELEC 50128. Since none of the tools compared cover the full V-Cycle on its own, combinations of tools are required. Even though there are no requirements specified with respect to support for the V-cycle phases compares the tools with respect to the support provided for V-cycle phases.

Number	V-Cycle Phase and SIL 4 coverage	Scade toolchain	B/Event B toolchain	SPARK toolchain
1	System & Requirement Analysis	3	9	0
2	Architecture Design (semi-formal)	9	6	0
3	Software Design (fully-formal)	9	9	9
4	Software Code Generation	9	9	9
5	CENELEC 50128 Certification for SIL 4	9	9	9

Table 5 - Comparison of support for V-cycle phases

4 SWOT Analysis

This section provides a SWOT analysis of the Model-based Software Development tools compared.

4.1 SWOT Analysis for SCADE

Strengths:

The biggest strength of SCADE by far is that it works “out of the box”, with barely any tailoring required. SCADE has been developed for embedded system and software engineering, and this is exactly the right field of application. The system engineering part is based on the open source Papyrus technology that has been customized and streamlined, so little work needs be done here. Finally, SCADE supports all phases of the V-Cycle according to CENELEC EN 50128.

Weaknesses:

SCADE is not open source and would not meet any open source objectives. However, the producer of SCADE has offered to DB to open their technology to select customers in the railway domain if it is of strategic importance to them. Nevertheless, this needs to be negotiated and is still an open topic.

Opportunities:

Using SCADE would doubtlessly increase the chances of success of modeling activities. By using SCADE, we would ensure a model-based approach covering all necessary aspects of the V-cycle of CENELEC EN 50128.

There is another opportunity: by dangling the chance to adapt SCADE and potentially opening a large market segment, ANSYS (manufacturer of SCADE) may decide to open SCADE.

Threats:

There is the real threat that ANSYS is encouraging us to adapt SCADE under the premise of opening parts of SCADE to meet our requirements. If opening of SCADE would be necessary, we must analyse this issue soon.

4.2 SWOT Analysis for B-/Event B Toolchain

Strengths:

B/Event-B provides a single framework for formal development and verification, from system specification to software implementation.

B/Event-B is proven in the rail industry, where it has been deployed successfully. There is also a strong body of academic research that can be taken advantage of.

There are tools available, both open source and commercial.

- Atelier B platform *(http://www.atelierb.eu/)* provides tools supporting system and software formal development with B/Event B. Although not completely open source, the type-checker being the only exception, it is a free-of-charge platform with a full-time dedicated support team and it is runnable on common PCs.
- Rodin platform (*http://www.event-b.org/)* provides tools supporting system formal development with Event-B and allowing moving from an UML/state-machine based system specification to a B/Event-B formal model. It is an Eclipse-based and open source platform runnable on common PCs.
- ProB *(https://www3.hhu.de/stups/prob/index.php/Main_Page)* is a formal verification and animation tool of B/Event-B models. ProB can be used either standalone or integrated in Atelier B and Rodin platforms. It is open source and is runnable on common PCs.

There is B-related expertise in the consortium, ensuring that questions can be answered and that there is a commercial incentive.

Weaknesses:

Rodin and ProB are academic tools without a full-time dedicated support team.

While B/Event-B is well-suited for modelling state-based systems, it is more difficult to model real-time behaviour.

Opportunities:

B/Event-B could be a tradeoff between using a commercially proven approach (like SCADE), while still residing in the open source realm (like TOPCASED). And as there are more options available, both open and closed, the risk is lower.

Threats:

Acceptance may be the biggest problem, as B/Event-B is a “hard core” formal notation, which is considered hard to read. We must be prepared to train the users and ensure that they accept it beforehand.

4.3 SWOT Analysis for SPARK Toolchain

Strengths:

SPARK is a subset of ADA with additional notations to express formal contracts and to do formal verification *(http://www.spark-2014.org/)*. SPARK is a language and the AdaCore SPARK Pro tool will formally verify the SPARK code *(https://www.adacore.com/sparkpro)*. SPARK Discovery is open source version of the AdaCore SPARK Pro tool (https://www.adacore.com/download). SPARK is certified according to CENELEC 50128, the use of the tool within the EN 50128 framework is documented (cf. *https://www.adacore.com/books/cenelec-en-50128-2011*) and is already used in safety critical domains e.g. aerospace, and in the security domain.

Weaknesses:

Compared to SCADE: SPARK will support the real expression of formal contracts and do formal verification of them. Data types are more elaborated (e.g. variant records, integer range, fixed point …).

Compared to B/Event-B: SPARK is closer to common development language e.g. C and therefore more user friendly.

Compared to both SCADE and B/Event-B: SPARK is a standardized language as part of International Standard ISO/IEC 8652:2012(E) (aka Ada 2012). All Ada documentation is freely available on the web (http://www.ada-auth.org/standards/ada12.html).

SPARK support direct testing and simulation like testing ADA or C-Software. Using SPARK, one can use regular testing complementary to formal verification. Depending on the use case sometimes it is easier to do regular testing than formal verification. An advantage is also that one can mix formal verification and regular testing.

Opportunities:

SPARK is open source and is easier to understand than the B-Method since it is closer to a common software development language. Therefore, SPARK may achieve a high open source community interest for safety critical software. It covers a full open source toolchain part for the software design phase in the V-Model.

Threats:

Compare to B/Event-B: it will very difficult to verify the high level functional software properties because SPARK lacks refinement capabilities of B/Event-B.

4.4 Summary

This section provides a summary of the three different tool options compared.

4.4.1 Option 1: SCADE-based Toolchain

Of course, SCADE in combination with PTC Modeler and Doors is maybe the most available toolchain which covers the full V-Cycle, since it comes also with a safety analysis tool. SCADE is also supporting a formal model-driven approach and therefore already in use for different domains. This leads also to the fact, that SCADE is easy to understand for railway systems engineers. However, we it must be mentioned that SCADE is a commercial tool and therefore has higher cost (licence fees) compared to Option 2 or Option 3.

4.4.2 Option 2: B/Event B-based Toolchain

This toolchain will with the combination of the Pro-B tool and the Event B method deliver full coverage of the V-Cycle phases. Of course, the Pro-B tool needs some development and therefore investment, since it is still on an academic level. Also, the simulation and optimized code generation will be weak within this toolchain. Furthermore, for the safety analysis an additional tool is necessary, but this is available on the market. A huge benefit of this toolchain is the fact that all parts will support an open source view and therefore the long-term support.

4.4.3 Option 3: SPARK-based Toolchain

SPARK is a very interesting option, since SPARK is a subset of Ada and supports formal verification. It is not a model-driven method, but closer to common programming languages and therefore easier to understand for railway systems engineers. SPARK lacks a safety analysis tool just as the tool chain for the B-Method. Furthermore, SPARK needs either IBM Rhapsody or PTC Modeler to cover the full V-Cycle. Therefore, some effort and investment are required to ensure the integration, traceability and support for IBM Rhapsody and PTC Modeler.

However, SPARK is open source, and a very interesting alternative to Option 1 and Option 2 with a huge opportunity to get an open source community attention. Also, one must consider the necessary investment to interface between higher level tools and the lower-level SPARK tool.

5 Overview Tools

Tool Name	Purpose
MsOffice (Word, Excel, PowerPoint, Project)	General Purpose
Webex / MsTeams	Telco
Public Repository	https://github.com/OCORA-Public
Internal Repository	https://github.com/openETCS/OCORA
Polarion	Requirement Engineering and Management
Capella	Model-based Systems Engineering
SCADE	Model-based Software Development

Table 6 - Overview Tools

Additional tools (ex. Use-Cases, Testing) might be listed at a later stage once the OCORA collaboration reaches later phases.

6 Internal Repository: GitHub

The OCORA internal repository can be found under: https://github.com/openETCS/OCORA

After each user is registered individually in GitHub. The OCORA Repository Administrator (baseliyos.jacob@deutschebahn.com or rolf.muehlemann2@sbb.ch) can grant access once the user name is provided. Github as platform offers a number of additional functions which might be used at a later stage.

7 Public Repository: GitHub

The OCORA public repository can be found under: https://github.com/OCORA-Public
A user registration is only required in case of commenting.

8 Requirement Engineering and Management: Polarion

Polarion is a requirement management tool certified to allow safety relevant developments. It ensures required functionality such as collaboration, traceability and workflow to pass validation. Polarion has become a de facto standard in safety relevant requirement engineering and management.

9 Model-based Systems Engineering: Capella

Capella is an open source software package for MBSE, supporting the Arcadia method. Hosted at polarsys.org, this solution provides a process and tooling for graphical modelling of systems, hardware or software architectures, in accordance with the principles and recommendations defined by the Arcadia method. Capella is mainly used for modelling complex and safety-critical systems in embedded systems development for industries such as aerospace, avionics, transportation, space, communications and security and automotive. For the following reasons, OCORA has decided to use Capella for MBSE:

- Capella is a dedicated, powerful tool to support the Arcadia method
- Most founding members of OCORA are using Capella in their CCS projects already

It is yet to be decided to what extent and in what phases of the product definition/development cycle Capella will be used.

10 Model-based Software Development: Scade

ANSYS SCADE Suite is a model-based development environment for critical embedded software. With native integration of the formally defined SCADE language, SCADE Suite is the integrated design environment for critical applications including requirements management, model-based design, simulation, verification, qualifiable/certified code generation and interoperability with other development tools and platforms.

SCADE Suite is used to design critical software, such as rail interlocking systems and signalling, automatic train operation, computer based train control, emergency braking systems, overspeed protection, train vacancy detection and many other aerospace, railway, energy, automotive and industrial applications.