Ref. Ares(2025)8485350 - 07/10/2025

FEDERATE

# Software-Defined Vehicle Support and Coordination Project

## D3.2 Second Prioritized Backlog Report

Authors: Daniel Krippner, Detlef Zerfowski

September 2025

| Deliverable | D3.2 – Second Prioritized Backlog Report |
|---|---|
| **Work Package(s)** | WP2 - Technology and high-level requirements solicitation<br>WP3 - Common Understanding |
| **Dissemination Level** | public |
| **Due Date** | 30-09-2025 |
| **Actual Submission Date** | 07-10-2025 |
| **WP Leader** | Detlef Zerfowski (ETAS GmbH) |
| **Deliverable Leader** | Detlef Zerfowski (ETAS GmbH) |
| **Contact Person** | Detlef Zerfowski (ETAS GmbH), Daniel Krippner (ETAS GmbH) |
| **Email** | Detlef.zerfowski@etas.com, Daniel.Krippner@etas.com |

| Document History | | | |
|---|---|---|---|
| **Revision No.** | **Date of the review** | **Name of the reviewer** | **Status of the document** ***(in progress, ready for review, released)*** |
| V0.1 | 06-10-2025 | Peter Priller, AVL | Revision of the first draft |
| V1.0 | 07-10-2025 | Claudia Keinrath, AVL | Final Version |

This document and the information contained within may not be copied, used, or disclosed, entirely or partially, outside of the **FEDERATE** consortium without prior permission of the project partners in written form

The project has been accepted for funding within the Chips Joint Undertaking (CHIPS JU), a public-private partnership in collaboration with the Horizon Europe (HORIZON) Framework Programme under Grant Agreement No. **101139749**

*Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Commission. Neither the European Union nor the granting authority can be held responsible for them.*

# 1 Table of Contents

*2 Executive Summary ..... 5*

*3 Introduction ..... 6*

3.1 Introduction - Purpose of this document ..... 6

3.2 Aims and Objectives of the Deliverable ..... 6

3.3 Context and Fit within the Project ..... 6

*4 Enhancement and Streamlining of Open Source Software Building Blocks: A Strategic Contribution to the FEDERATE Initiative ..... 7*

4.1 Executive Summary ..... 7

4.2 Project Context and Strategic Importance ..... 7

4.2.1 The FEDERATE Initiative Framework ..... 7

4.2.2 Repository Mission and Architectural Vision ..... 7

4.3 Technical Analysis of Pull Request #80 ..... 7

4.3.1 Scope and Scale of Contributions ..... 7

4.3.2 Strategic Focus on Open Source Software Building Blocks ..... 8

4.4 Impact Assessment and Strategic Implications ..... 8

4.4.1 Immediate Technical Benefits ..... 8

4.4.2 Community and Ecosystem Impact ..... 9

4.5 Future Directions and Recommendations ..... 9

4.5.1 Continued Evolution and Maintenance ..... 9

4.5.2 Strategic Recommendations for Stakeholders ..... 9

4.6 Conclusion ..... 10

*5 Update Building Block Catalog Snapshot ..... 11*

## Tables

Table 1 - FEDERATE BB Catalog Snapshot ..... 11

## 2 Executive Summary

The FEDERATE project, aimed at advancing the development of software-defined vehicle (SDV) components, has continued to make progress in its second year. Focus shifted to completing the collection and improving the organization of SDV building block descriptions. WP2 and WP3 played an important role in several key areas, contributing to the project's success, and ensuring that foundational elements are effectively established.

A primary goal of the project was the improvement of the FEDERATE GitHub repository, which houses an elaborate folder structure as organizing principle for SDV software building blocks. Among the new features added are export scripts that generate text- or excel representations of the building block library.

This document describes the building block updates and extensions, to exemplify the work of the project team. It is strongly connected with D3.4 (First orchestrated backlog report)

The workflow for creating and reviewing issues on the platform was exercised and streamlined, ensuring smooth collaboration among project participants. The modus operandi is explained in section 4, based on pull-request #80. WP2 and WP3 hosted discussions around the organizational structure of the SDV building blocks, providing feedback and insights that evolved the collection.

# 3 Introduction

## 3.1 Introduction - Purpose of this document

The purpose of this document is to provide an account of the contributions generated by WP2 and WP3 within the scope of the FEDERATE project. It aims to outline the key contributions, methodologies, and outcomes, particularly focusing on the creation and organization of the SDV software building blocks repository. By providing a comprehensive overview of these activities, the report ensures transparency, facilitates knowledge sharing among project stakeholders, and supports ongoing and future work within the FEDERATE project by offering a clear reference point for the project's progress and achievements thus far.

## 3.2 Aims and Objectives of the Deliverable

The primary aim of the deliverable is to describe the repository of SDV software building blocks (BB), which can serve as reference catalogue for developers and integrators of software-defined vehicle components. The repository is designed to collect and organize BB descriptions and use cases, both for existing BB's and BB's, enabling the project team to systematically represent the development of SDV software components. The objective is to provide a robust collection that facilitates collaboration among project participants, supports the efficient creation and review of building blocks, and ensures that the SDV components developed can be in line with industry needs, standards, and best practices.
The modus operandi is explained in the following section based on pull-request #80.

## 3.3 Context and Fit within the Project

This deliverable fits into the broader context of the FEDERATE project by addressing one of its core goals: the systematic development of SDV software components that can be utilized in the automotive industry. The repository serves as a central hub for organizing the building blocks that will form the basis of these components, ensuring that relevant BB descriptions and use cases are documented and accessible to the community, as well as to organizations that want to contribute to building block implementations. By providing a clear and structured framework, this deliverable ensures that subsequent development activities are grounded in well-defined and widely reviewed inputs.

# 4 Exemplary interaction with the Proposed Building Block Repository

## 4.1 Executive Summary

This chapter presents a comprehensive analysis of Pull Request #80, titled "Add and streamline OSS Building Blocks," submitted to the CSA-FEDERATE Proposed-BuildingBlocks repository. This significant contribution represents an example on how to interact with the repository, to support the SDV ecosystem development initiative. The pull request, encompassing substantial additions of references to existing open-source project building blocks, demonstrates a methodical approach to expanding and refining the FEDERATE Proposed Building Block Repository.

## 4.2 Project Context and Strategic Importance

### 4.2.1 The FEDERATE Initiative Framework

FEDERATE focusses on creating a comprehensive ecosystem of non-differentiating, reusable software building blocks that serve as the foundational infrastructure for SDV development, deployment, and evolution. The establishment of standardized, open-source building blocks becomes crucial for maintaining European competitiveness while fostering innovation and collaboration across the continental automotive ecosystem.

### 4.2.2 Repository Mission and Architectural Vision

The CSA-FEDERATE/Proposed-BuildingBlocks repository serves as the central reference library for cataloging, organizing, and disseminating structured descriptions of Building Blocks (BBs) essential for SDV development. This repository transcends traditional code repositories by providing a structured reference library of (top-level) architectural specifications and open source projects rather than executable code. This strategic approach enables a broader stakeholder community to contribute to the standardization and evolution of SDV technologies.

The repository facilitates knowledge sharing across diverse automotive stakeholders, including original equipment manufacturers (OEMs), tier-one suppliers, software developers, academic institutions, and regulatory bodies. By providing standardized templates, conventions, and collaborative mechanisms, the repository ensures that SDV-relevant ideas and technical components can be effectively communicated, evaluated, and integrated into the broader European automotive ecosystem.

## 4.3 Technical Analysis of Pull Request #80

### 4.3.1 Scope and Scale of Contributions

Pull Request #80 (https://github.com/CSA-FEDERATE/Proposed-BuildingBlocks/pull/80) is representative for a more substantial single contribution to the FEDERATE Building Blocks repository, with its scope of 6,826 line additions balanced against strategic 25-line removals. This expansion indicates a comprehensive effort to enhance the repository's coverage of open-source software building blocks critical to SDV development. The nature of these additions suggests the incorporation of multiple new building block categories, detailed

specifications, implementation guidelines, and collaborative frameworks.

The balance between additions and removals demonstrates a thoughtful approach to repository maintenance, where obsolete or redundant content is eliminated while new, valuable contributions are integrated. This methodology ensures that the repository maintains its relevance and utility while preventing information bloat that could hinder usability and navigation.

### 4.3.2 Strategic Focus on Open Source Software Building Blocks

The pull request's emphasis on Open Source Software (OSS) building blocks aligns perfectly with the FEDERATE initiative's commitment to creating accessible, collaborative, and non-proprietary solutions for the European automotive industry. By focusing on OSS components, the contribution supports several strategic objectives:

- **Accessibility and Democratization**: OSS building blocks ensure that smaller automotive companies, startups, and academic institutions can participate meaningfully in the SDV ecosystem without prohibitive licensing costs or proprietary restrictions. This democratization of access is crucial for maintaining Europe's innovative edge in automotive technologies.
- **Transparency and Trust**: Open-source components provide complete visibility into their implementation, security characteristics, and performance attributes. This transparency is essential for safety-critical automotive applications where regulatory compliance and public trust are paramount.
- **Collaborative Innovation**: The open-source model facilitates collaborative development, where improvements and innovations can be rapidly shared across the entire European automotive ecosystem. This collaborative approach accelerates innovation cycles and ensures that best practices are quickly disseminated.
- **Vendor Independence**: By emphasizing OSS solutions, the FEDERATE initiative reduces European dependence on proprietary technologies from non-European vendors, supporting strategic autonomy in critical automotive technologies.

## 4.4 Impact Assessment and Strategic Implications

### 4.4.1 Immediate Technical Benefits

The implementation of Pull Request #80 delivers several immediate technical benefits to the FEDERATE ecosystem:

- **Expanded Coverage**: The substantial additions significantly broaden the repository's coverage of OSS building blocks, providing SDV developers with a more comprehensive toolkit for system development.
- **Improved Accessibility**: Streamlining efforts make the repository more user-friendly, reducing the time and effort required for stakeholders to identify and implement relevant building blocks.
- **Enhanced Collaboration**: Better organization and documentation facilitate more effective collaboration among diverse stakeholders in the European automotive ecosystem.
- **Quality Assurance**: The careful balance of additions and removals suggests a quality-focused approach that maintains high standards while expanding capabilities.
- Beyond immediate technical benefits, Pull Request #80 contributes to several long-term strategic objectives:
    - **Ecosystem Maturation**: The comprehensive nature of the contribution supports the

maturation of the European SDV ecosystem by providing more complete BB catalogue.
- **Innovation Acceleration**: Standardized OSS building blocks enable faster innovation cycles by providing proven, reliable components that can be rapidly integrated into new SDV solutions.
- **Market Positioning**: The enhanced repository strengthens Europe's position in the global SDV market by providing superior collaborative infrastructure and comprehensive technical resources.
- **Regulatory Readiness**: Well-documented, standardized building blocks facilitate regulatory compliance and approval processes, which are crucial for automotive applications.

### 4.4.2 Community and Ecosystem Impact

The pull request's focus on OSS building blocks and streamlined organization benefits the broader FEDERATE community:

- **Increased Participation**: Improved accessibility and organization lower barriers to entry for new contributors, potentially expanding the active community around the FEDERATE initiative.
- **Knowledge Transfer**: Better documentation and organization facilitate knowledge transfer between organizations and individuals, supporting the overall growth of SDV expertise across Europe.
- **Quality Improvement**: The streamlined approach establishes best practices for future contributions, raising the overall quality bar for the repository.
- **Innovation Culture**: The emphasis on open-source solutions and collaborative development reinforces the innovation culture that is essential for maintaining European leadership in automotive technologies.

## 4.5 Future Directions and Recommendations

### 4.5.1 Continued Evolution and Maintenance

The successful implementation of Pull Request #80 establishes a foundation for continued evolution of the FEDERATE Building Blocks repository. Future efforts should focus on:

- **Community Engagement**: Leveraging the improved accessibility and organization to attract broader community participation and contribution.
- **Validation and Testing**: Developing systematic approaches to validate the effectiveness and completeness of the building blocks in real-world SDV development scenarios.
- **Cross-Industry Collaboration**: Extending the collaborative model to include stakeholders from adjacent industries such as telecommunications, cloud computing, and cybersecurity.
- **International Coordination**: Exploring opportunities for collaboration with similar initiatives in other regions while maintaining European strategic autonomy.

### 4.5.2 Strategic Recommendations for Stakeholders

Organizations participating in or benefiting from the FEDERATE initiative should consider several strategic actions based on the advancements represented by Pull Request #80:

- **Active Participation**: Engage actively in the repository's evolution by contributing building blocks, providing feedback, and participating in collaborative development processes.

- **Implementation Planning**: Develop systematic approaches to evaluate and implement relevant building blocks in their SDV development activities.
- **Knowledge Investment**: Invest in developing internal expertise in the building blocks and collaborative methodologies represented in the repository.
- **Partnership Development**: Explore partnerships with other FEDERATE stakeholders to leverage the expanded building block ecosystem more effectively.

## 4.6 Conclusion

Pull Request #80 represents an exemplarily contribution to the FEDERATE building block collection, demonstrating the potential for collaborative, open-source approaches to address the complex challenges of Software-Defined Vehicle development. The substantial expansion of OSS building blocks, combined with systematic streamlining efforts, establishes a stronger foundation for European leadership in automotive technologies.

The strategic importance of this contribution extends beyond its immediate technical benefits to encompass broader objectives of ecosystem development, innovation acceleration, and competitive positioning. By emphasizing open-source solutions and collaborative development, Pull Request #80 reinforces the values and methodologies that are essential for the success of the FEDERATE initiative.

The successful implementation of this contribution provides a model for future development efforts and demonstrates the effectiveness of the collaborative approach embodied in the FEDERATE project.

The ongoing evolution of the FEDERATE Building Blocks repository, exemplified by contributions such as Pull Request #80, represents a critical investment in the future of European automotive technology. By providing comprehensive, accessible, and high-quality building blocks for SDV development, the FEDERATE initiative is positioning Europe to lead the global transition to software-defined mobility solutions.

This contribution stands as a testament to the power of collaborative development and open-source innovation in addressing complex, industry-wide challenges. The methodical approach to expansion and streamlining demonstrated in Pull Request #80 establishes best practices that will guide future development efforts and ensure the continued evolution of the FEDERATE ecosystem.

# 5 Update Building Block Catalog Snapshot

The current set of Building Blocks is presented via Table 1, including structural metadata (BB Tags and Layer) and implementation project URLs (where applicable). The full catalog repository is located at https://github.com/CSA-FEDERATE/Proposed-BuildingBlocks

*Table 1 - FEDERATE BB Catalog Snapshot*

| BB Name | Implementation Status | BB Tags(s) | Layer | Known Implementation |
|---|---|---|---|---|
| quantumSAR | implementation exists | BB-SC | MWLayer | https://github.com/iavofficial/IAV_quantumSAR |
| Eclipse Bluechi | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-bluechi/bluechi |
| Eclipse Muto | implementation exists | BB-SC | MWLayer | https://eclipse-muto.github.io/docs/ |
| Eclipse Pullpiri Service Orchestrator | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-pullpiri/pullpiri |
| Eclipse S-CORE FEO | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-score/feo |
| Eclipse OpenSOVD | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-opensovd/opensovd |
| COVESA Open1722 | implementation exists | BB-SC | MWLayer | https://github.com/COVESA/Open1722 |
| Eclipse KUKSA | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-kuksa |
| COVESA VISSR | implementation exists | BB-SC | MWLayer | https://github.com/COVESA/vissr |
| Eclipse S-CORE LOLA | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-score/communication |
| Eclipse Zenoh | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-zenoh/zenoh |
| COVESA uServices | implementation exists | BB-SC | MWLayer | https://github.com/COVESA/uservices |
| Eclipse eCAL | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-ecal/ecal |
| Eclipse Canaught | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-canought |
| XCPlite | implementation exists | BB-SC | MWLayer | https://github.com/vectorgrp/XCPlite |
| Eclipse uProtocol | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-uprotocol |
| Eclipse iceoryx2 | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-iceoryx/iceoryx2 |
| OpenDDS | implementation exists | BB-SC; BB-MU; BB-CSC; BB-CMU | MWLayer | https://github.com/OpenDDS/OpenDDS |
| CycloneDDS | implementation exists | BB-SC; BB-MU; BB-CSC; BB-CMU | MWLayer | https://github.com/eclipse-cyclonedds/cyclonedds |
| Fast-DDS | implementation exists | BB-SC; BB-MU; BB-CSC; BB-CMU | MWLayer | https://github.com/eProsima/Fast-DDS |

| | | | | |
|---|---|---|---|---|
| vSOME/IP | implementation exists | BB-SC; BB-MU; BB-CSC; BB-CMU | MWLayer | [vSomeIP](https://github.com/COVESA/vsomeip) |
| embeddedRTPS - DDS for Ressource-constrained environments | implementation exists | BB-SC, BB-CSC, BB-MU, BB-CMU | MWLayer | Github repo: <https://github.com/embedded-software-laboratory/embeddedRTPS> |
| Eclipse Ankaios | implementation exists | BB-SC | MWLayer | https://github.com/eclipse-ankaios/ankaios |
| Eclipse Symphony Agents | implementation exists | BB-SC | MWLayer | https://eclipse-symphony.github.io/symphony-website/ |
| Eclipse Heimlig | implementation exists | BB-SC | OSLayer | https://github.com/eclipse-heimlig/heimlig |
| XENProject | implementation exists | BB-SC; BB-CSC | HWLayer | https://github.com/xen-project/xen |
| HaloOS | implementation exists | BB-SC, BB-EST | MWLayer, OSLayer | https://gitee.com/haloos |
| OpenHarmony | implementation exists | BB-SC, BB-EST | MWLayer, OSLayer | https://gitee.com/openharmony |
| Eclipse S-CORE | implementation exists | BB-SC, BB-EST | MWLayer, OSLayer | https://eclipse-score.github.io/ |
| Autoware | implementation exists | BB-SC, BB-EST | MWLayer, OSLayer | https://github.com/autowarefoundation/autoware |
| Eclipse Leda | implementation exists | BB-EST | | https://github.com/eclipse-leda |
| L4Re | implementation exists | BB-SC, BB-EST | MWLayer, OSLayer | https://github.com/kernkonzept/manifest |
| AGL Automotive Grade Linux | implementation exists | BB-SC, BB-EST, BB-EST-TC | APPLayer, MWLayer, OSLayer | https://github.com/orgs/agl-ic-eg/repositories?type=all |
| Android Automotive | implementation exists | BB-SC, BB-EST, BB-EST-TC | APPLayer, MWLayer, OSLayer | https://source.android.com/docs/automotive |
| Eclipse ThreadX | implementation exists | BB-SC, BB-EST | OSLayer | https://github.com/eclipse-threadx/rtos-docs |
| Linux Foundation Zephyr | implementation exists | BB-SC, BB-EST | OSLayer | https://github.com/zephyrproject-rtos |
| seL4 | implementation exists | BB-SC, BB-EST | OSLayer | https://github.com/seL4 |
| Eclipse OpenBSW | implementation exists | BB-SC, BB-EST | MWLayer, OSLayer | https://github.com/eclipse-openbsw |
| Apache Foundation NuttX | implementation exists | BB-SC, BB-EST | OSLayer | https://github.com/apache/nuttx |

| | | | | |
|---|---|---|---|---|
| Private Cross OEM Joint Compute for EV Charging | implementation exists | S-BB | MWLayer | https://covesa.global/project/private-cross-oem-joint-compute-for-ev-charging/ |
| Data Architecture Terminology | implementation exists | S-BB | MWLayer | https://covesa.global/project/data-architecture-terminology/ |
| VSS | implementation exists | S-BB | MWLayer | https://github.com/covesa/vehicle_signal_specification |
| Unified Push Notification | implementation exists | S-BB | MWLayer | https://covesa.global/project/unified-push-notifications/ |
| HIM | implementation exists | S-BB | MWLayer | https://github.com/COVESA/hierarchical_information_model |
| InCar Wallet Payments and Orchestration | implementation exists | S-BB | MWLayer | https://covesa.global/project/in-car-wallet-payments-orchestration/ |
| EV Charging Event Data Aggregation | implementation exists | S-BB | MWLayer | https://covesa.global/project/ev-charging-event-data-aggregation/ |
| VSS | implementation exists | S-BB | MWLayer | https://covesa.global/project/common-vehicle-capabilities/ |
| eSync Alliance | implementation exists | S-BB | MWLayer | https://esyncalliance.org/technology/ |
| Pilot Auto Reference Design | implementation exists | S-BB | MWLayer | https://docs.pilot.auto/en/reference-design/ |
| VISS | implementation exists | S-BB | MWLayer | https://github.com/COVESA/vehicle-information-service-specification |
| Linux Foundation EVerest | implementation exists | S-BB | MWLayer | https://everest.github.io/nightly/ |
| Commercial Vehicle Information Specifications | implementation exists | S-BB | | https://github.com/COVESA/commercial-vehicle-information-specifications |
| Eclipse AutoWRX | implementation exists | BB-CSC-TC, BB-SC-TC, BB-CEST, BB-EST | APPLayer, MWLayer | https://github.com/eclipse-autowrx |
| Eclipse Connected Services Platform | implementation exists | BB-CSC, BB-CSC-TC, BB-CEST, BB-SC | APPLayer | https://eclipse-ecsp.github.io/ecsp-website/ |
| Lichtblick | implementation exists | BB-CEST, BB-CSC-TC | AppLayer | https://github.com/Lichtblick-Suite/lichtblick |
| Eclipse Cloe | implementation exists | BB-CEST, BB-CSC-TC | AppLayer | https://cloe.readthedocs.io/en/latest/ |

| | | | | |
|---|---|---|---|---|
| SODA.Sim | implementation exists | BB-CEST, BB-CSC-TC | AppLayer | https://github.com/soda-auto/soda-sim |
| OSTAR | implementation exists | BB-CEST, BB-CSC-TC | AppLayer | https://github.com/DLR-TS/OSTAR-Quickstart |
| Eclipse SCM | implementation exists | BB-CEST, BB-CSC-TC | AppLayer | https://gitlab.eclipse.org/eclipse/scm/scm |
| Eclipse Environment Simulator Minimalistic (esmini) | implementation exists | BB-CEST, BB-CSC-TC | AppLayer | https://eclipse.dev/sumo/ |
| Eclipse Mosaic | implementation exists | BB-CEST, BB-CSC-TC | AppLayer | https://github.com/eclipse-mosaic/mosaic |
| Eclipse openPASS | implementation exists | BB-CEST, BB-CSC-TC | AppLayer | https://openpass.eclipse.org/ |
| Eclipse SUMO | implementation exists | BB-CEST, BB-CSC-TC | AppLayer | https://eclipse.dev/sumo/ |
| Eclipse Symphony Orchestration Layer | implementation exists | BB-CEST, BB-CSC-TC, BB-SC | ## BB Tags(s) BB-CEST, BB-CSC-TC, BB-SC ## Functional Clusters ## Layer MWLayer | https://eclipse-symphony.github.io/symphony-website/ |
| Eclipse Pullpiri Package Service | implementation exists | BB-CEST, BB-CSC-TC | AppLayer | https://github.com/eclipse-pullpiri/pullpiri |
| Eclipse Ankaios Dashboard | implementation exists | BB-CEST, BB-CSC-TC | MWLayer | https://github.com/eclipse-ankaios-dashboard/ankaios-dashboard |
| Eclipse Muto Dashboard | implementation exists | BB-CEST, BB-CSC-TC | MWLayer | https://github.com/eclipse-muto/dashboard-device |
| Eclipse Tractus-X | implementation exists | BB-CEST | APPLayer | https://github.com/eclipse-tractusx |

| | | | | |
|---|---|---|---|---|
| SKyBT | implementation exists | BB-EST, BB-CEST, BB-SC-TC, BB-CSC-TC | MWLayer | Github repo: https://gitlab.eclipse.org/eclipse/skybt/skybt |
| VELOCITAS | implementation exists | BB-EST | | https://github.com/eclipse-velocitas |
| Yocto project | implementation exists | BB-EST, BB-CEST | | https://git.yoctoproject.org/ |
| OpenFastTrace | implementation exists | BB-EST | | https://github.com/itsallcode/openfasttrace |
| Sphinx Needs | implementation exists | BB-EST | | https://github.com/useblocks/sphinx-needs |
| IFEX | implementation exists | BB-EST | | https://github.com/COVESA/ifex |
| Eclipse OpenDUT | implementation exists | BB-EST, BB-CEST, BB-SC-TC, BB-CSC-TC | MWLayer | https://github.com/eclipse-opendut/opendut |
| Vector SIL Kit | implementation exists | BB-EST, BB-CEST, BB-SC-TC, BB-CSC-TC | | https://github.com/vectorgrp/sil-kit |
| Autowrx | implementation exists | BB-EST | | https://gitlab.eclipse.org/eclipse/autowrx/autowrx |
| OpenXilEnv | implementation exists | BB-EST | | https://github.com/eclipse-openxilenv/openxilenv |
| Central Data Service Playground | implementation exists | BB-EST | | https://github.com/COVESA/cdsp |