University of Stuttgart
Institute of Industrial Automation and
Software Engineering

FA <Thesis No.>

Flexible Sensor Data Abstraction in SDVs with Middleware-Driven Approach

<First Name> <Last Name>

<Program of Study>

User Requirements Specification

Examiner: Prof. Dr.-Ing. Michael Weyrich
Supervisor: Akshay Narla, M.Sc.

Document Version Management

Version	Author	QA	Date	Status	Changes
0.1	AN	MW		in progress	Creation

0 Table of Contents

1 Introduction to the Project

As the automotive industry advances through the CASE (Connected, Autonomous, Shared Services, Electrification) concepts, managing the complexity of Software-defined Vehicles (SDVs) becomes increasingly important. The SDV is expected to be always updateable (both software and hardware), connected and integrated into the consumer digital world to increase the value throughout the vehicle life-cycle. Software drives value creation in these future vehicles, serving functions and services within the vehicle, in the cloud, and in infrastructures around the vehicle, creating a connected mobility ecosystem. To meet these demands of the Software-defined Vehicle, the tightly coupled software and hardware in the current vehicle architecture need to be decoupled [1–3]. The complexity and heterogeneity observed in the current vehicle software architecture must be managed efficiently to provide an efficient software-defined environment. In this context, it is necessary to examine the concepts necessary in creating an open SDV platform along with standardized interfaces for sensors and actuators to allow software interoperability.

To overcome these challenges, this project focuses on employing Eclipse Kuksa, DDS (Data Distribution Service), and the Vehicle Signal Specification (VSS) to create a framework for sensor data abstraction that facilitates seamless data exchange and supports software interoperability. This framework seeks to abstract sensor-specific complexities and provide a vendor-neutral, standardized data access layer. By leveraging various open-source frameworks, the project aims to enable efficient integration of similar sensors (e.g., cameras) from different vendors into a cohesive software ecosystem. The inclusion of mixed-criticality considerations ensures that critical and non-critical software functions can run on shared hardware with appropriate resource isolation and fault tolerance. Ultimately, the project seeks to demonstrate how these technologies can simplify sensor data handling and pave the way for scalable and interoperable SDV architectures.

2 Objectives

This project aims to investigate how Eclipse Kuksa, DDS, and the Vehicle Signal Specification (VSS) can collaboratively enable sensor data abstraction for Software-Defined Vehicles (SDVs). The focus is creating a framework that provides vendor-neutral, standardized data interfaces to facilitate seamless data exchange between similar sensors (e.g., cameras) from different manufacturers. To achieve this, the project will explore mixed-criticality concepts to manage software of varying criticalities on shared hardware, ensuring robust resource isolation and efficient data handling. The project will demonstrate how these technologies can simplify sensor data abstraction, enhance interoperability, and provide a scalable and integrated SDV ecosystem by developing and implementing a proof-of-concept.

3 Operational Area

The findings of this literature work should be universally usable and accessible for the HAL4SDV project. The findings are intended to be non-confidential and publicly available.

4 Functional Requirements

/UFR10/ Literature Search

/UFR11/ The research shall review existing industry standards in sensor abstraction in an automotive context.

/UFR12/ The research shall review the features and capabilities of Eclipse Kuksa, DDS, and COVESA VSS in providing abstraction and standardization.

/UFR20/ Mixed-Criticality Interfaces

/UFR21/ The research shall investigate mixed-criticality concepts in automotive systems, where the same sensor hardware supports software with varying criticality levels.

/UFR22/ The research shall define requirements for middleware supporting mixed-criticality.

/UFR30/ Sensor Data Abstraction

/UFR31/ The project shall establish high-level requirements for sensor data abstraction using Kuksa.val and VSS

/UFR32/ The project shall design a conceptual framework integrating DDS for reliable communication and VSS for unified signal representation.

/UFR40/ Proof-of-Concept

/UFR41/ The project shall develop a simple prototype to demonstrate sensor data abstraction from two similar sensors (e.g., cameras) from different vendors.

/UFR42/ The project shall perform basic testing to evaluate interoperability, latency, and mixed-criticality performance.

5 Non-Functional Requirements

/UNR10/ Requirements for the Literature Search

/UNR11/ Papers should not be older than 5 years

/UNR12/ Basic literature can be older

/UNR20/ Proof of Concept shall demonstrate the concept as expected with appropriate evaluation methodology.

6 Quality Requirements

	very high	high	normal	not relevant
Theory		x		
Functionality		x		
Usability			x	
Changeability		x		
Transferability		x		

7 Execution

The thesis has to be executed according to the “IAS Process Model” (Open Model). In the following table, the documents which have to be created are listed.

Table 1: Documents to be created

Project Phase			**FA, MT**
PE 1 Definition Phase			
PE 1.1	Definition of Requirements	System Requirements Specification	X
PE 1.2		Intermediate Product 1.1	O
QA 1.2		→ Definition Review	X
PE 2 Project Phase 2			
PE 2.1		Intermediate Product 2.1	X
PE 2.2		Intermediate Product 2.2	O
QA 1.3		→ Review 2	X
PE 3 Project Phase 3			
PE 3.1		Intermediate Product 3.1	X
PE 3.2		Intermediate Product 3.2	O
QA 1.4		→ Review 2	X
PE 4 Project Phase 4			
PE 4.1		Intermediate Product 4.1	X
PE 4.2		Intermediate Product 4.2	O
QA 1.4		→ Acceptance Review	X

Legend:

X Document is mandatory
O Document is optional

The current state of the thesis and results have to be discussed with the tutor every two weeks.

The IAS guidelines have to be respected.

8 Additions

None.

9 References

[1] FEDERATE Consortium, *2024-04-12-SDVoF-Vision-document-ver017-final*.

[2] Z. Liu, W. Zhang, and F. Zhao, "Impact, Challenges and Prospect of Software-Defined Vehicles," *Automot. Innov.*, vol. 5, no. 2, pp. 180–194, 2022, doi: 10.1007/s42154-022-00179-z.

[3] G. Keßler, D. Sieben, A. Bhange, and E. Börner, "The Software Defined Vehicle – Technical and Organizational Challenges and Opportunities," in *23. Internationales Stuttgarter Symposium* (Proceedings), A. C. Kulzer, H.-C. Reuss, and A. Wagner, Eds., Wiesbaden: Springer Fachmedien Wiesbaden, 2023, pp. 414–426.

[4] "DDS Resources." Accessed: 25-Nov-24. [Online]. Available: https://www.dds-foundation.org/dds-resources/

[5] Neil Puthuff, *DDS: The Data Backbone of SDV Interoperability*.

[6] M. Pohnl, A. Tamisier, and T. Blass, "A Middleware Journey from Microcontrollers to Microprocessors," in *2022 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, Antwerp, Belgium, Mar. 2022 - Mar. 2022, pp. 282–286, doi: 10.23919/DATE54114.2022.9774609.

[7] A. Banijamali, P. Kuvaja, M. Oivo, and P. Jamshidi, "Kuksa$*$: Self-adaptive Microservices in Automotive Systems," in *Product-Focused Software Process Improvement* (Lecture Notes in Computer Science 12562), M. Morisio, M. Torchiano, and A. Jedlitschka, Eds., Cham: Springer International Publishing, 2020, pp. 367–384.

[8] GitHub. "eclipse/kuksa.val: kuksa.val." Accessed: 25-Nov-24. [Online]. Available: https://github.com/eclipse/kuksa.val

[9] P. H. M. Pereira, G. Cainelli, C. E. Pereira, J. P. J. Da Costa, and E. P. de Freitas, "An Interoperability Middleware for IIoT," in *2023 IEEE 32nd International Symposium on Industrial Electronics (ISIE)*, Helsinki, Finland, Jun. 2023 - Jun. 2023, pp. 1–6, doi: 10.1109/ISIE51358.2023.10227975.

[10] D. Slama, A. Nonnenmacher, and T. Irawan, *The software-defined vehicle : a digital-first approach to creating next-generation experiences / by Dirk Slama, Achim Nonnenmacher, and Thomas Irawan*. Sebastopol, CA: O'Reilly Media, Inc, 2023.

[11] C. Girish, "Evaluation and integration of DDS middleware for interconnection between Android Automotive and AUTOSAR," 2024, doi: 10.18419/OPUS-13970.[4–11]