

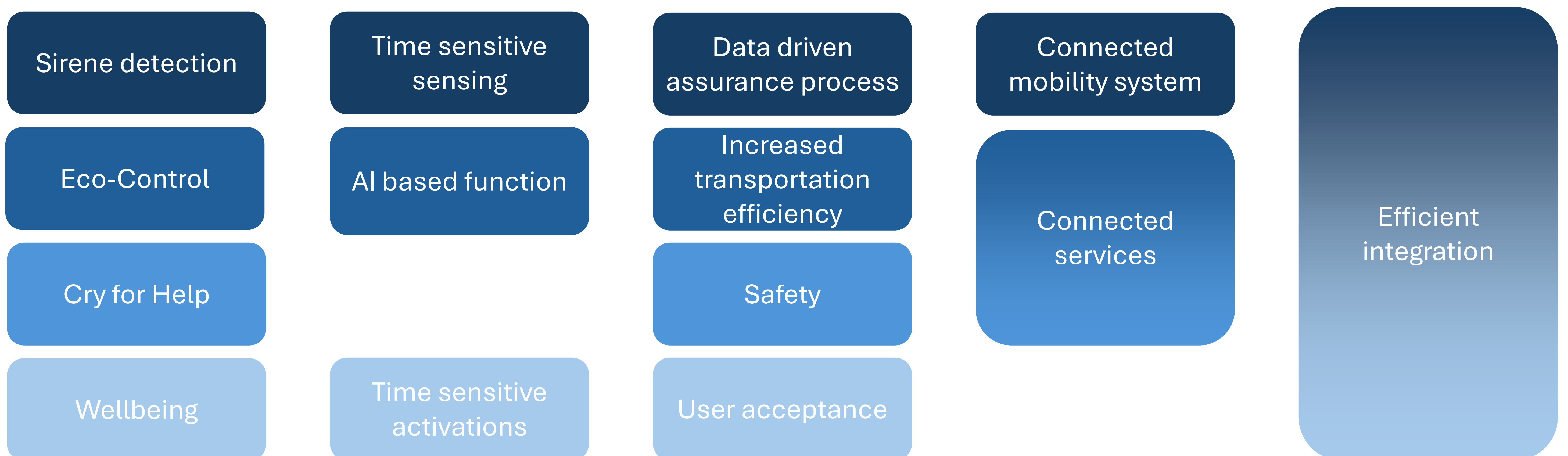
Use-Cases of SoAs

Integration and Safety Assurance Methodologies

With funding from the:



With various use cases for service-oriented architecture, we demonstrate the feasibility of novel methods to achieve platform-independent integration efficiency, data-driven assurance, and energy-efficient transportation. Data-centric applications, combined with service-oriented architectures, enable a reusable and scalable system through standardized interfaces. A key highlight is the interconnection of services across different vehicles with the cloud ecosystem, enabling a reactive and collaborative traffic system. Furthermore, user acceptance is enhanced through time-critical audio and visual countermeasures within a connected vehicle system.

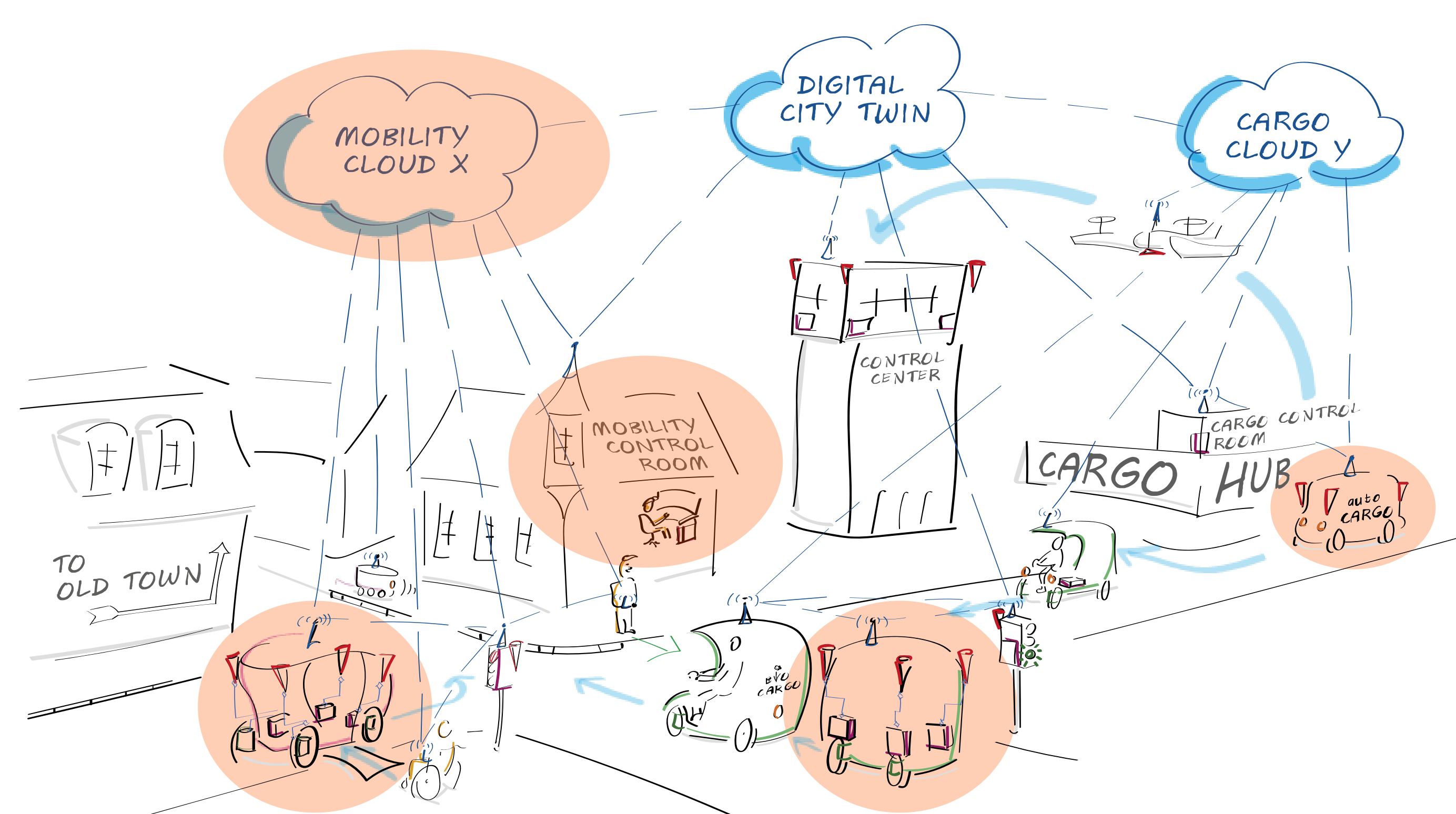


State of the Art

Vehicles are not yet reacting based on cloud data, and the potential for energy efficiency in transport systems is not fully leveraged through holistic cloud-based information. Current development processes are tied to individual frameworks that implement the AUTOSAR standard, but they are not prepared for integration into an open-source-oriented ecosystem.

Technological Innovations

- Responsive traffic ecosystem enabled by interconnected functions such as siren detection and cry-for-help recognition
- Efficient, platform-independent integration of service-oriented functions across various middleware frameworks
- Optimized workflow for configuring time-sensitive in-vehicle networks
- Assurance processes supported by methods and tools for audio-based AI functionalities
- Enhanced energy efficiency through optimized route calculation and execution



Contributing Partners



Universität Stuttgart



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Use-Cases of SoAs

Integration and Safety Assurance Methodologies

With funding from the:

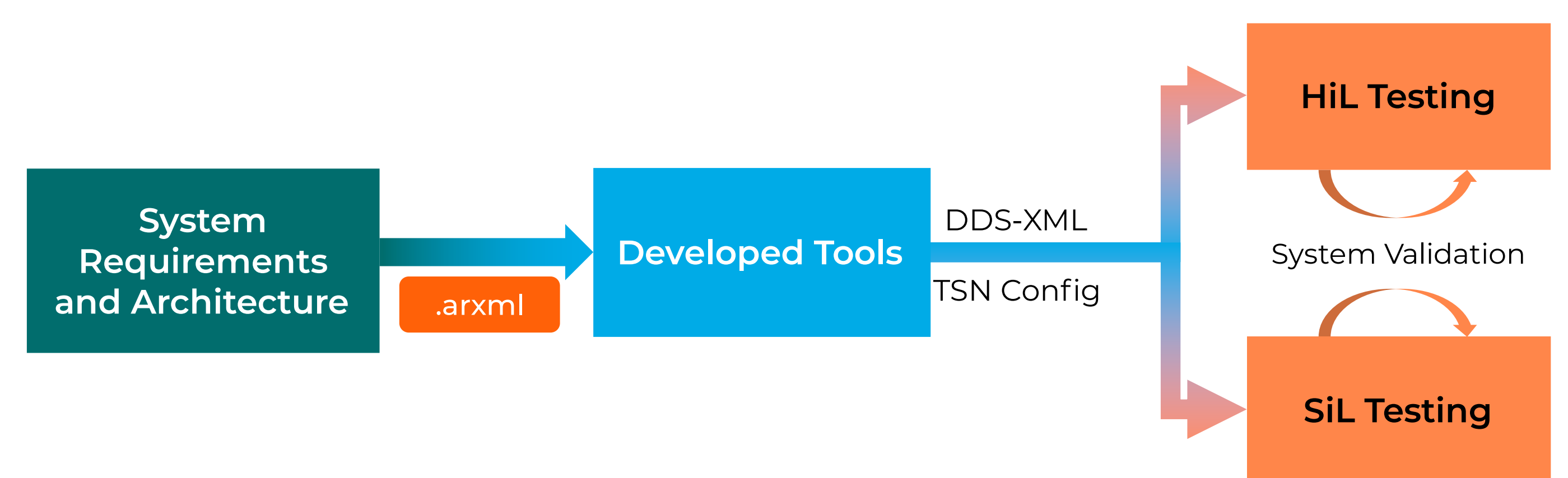
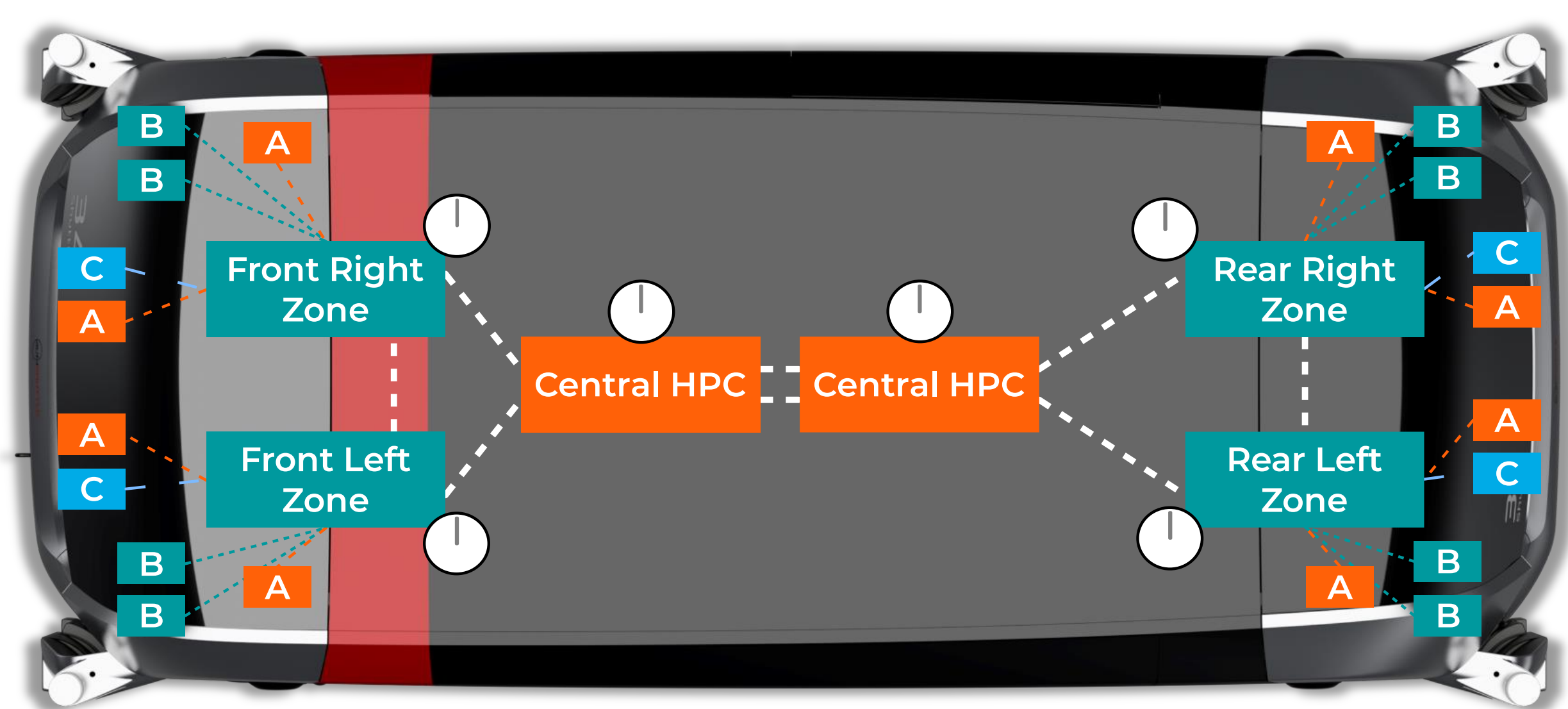


Objective

One objective of our project is to establish a process to combine the flexibility of Data Distribution Service (DDS) with the determinism of Time-Sensitive Networking (TSN) to enhance automotive communication systems. Specifically, we aim to achieve this within the context of a highly synchronized zonal architecture. By modeling the architecture and integrating these technologies, we seek to balance dynamic data distribution with strict timing requirements, ensuring reliable communication within automotive networks.

Workflow

Our workflow spans from system requirements to architectural modeling, which we perform using PREEvision. We develop a custom translator tool to extract the TSN and DDS configurations directly from the architectural model. Additionally, Network simulation configs are generated for future SiL and used to validate the setup, ensuring that the system performs as expected under various conditions. This automated process ensures consistency and accuracy in the configuration, facilitating seamless integration and validation of the system components.



Methodology

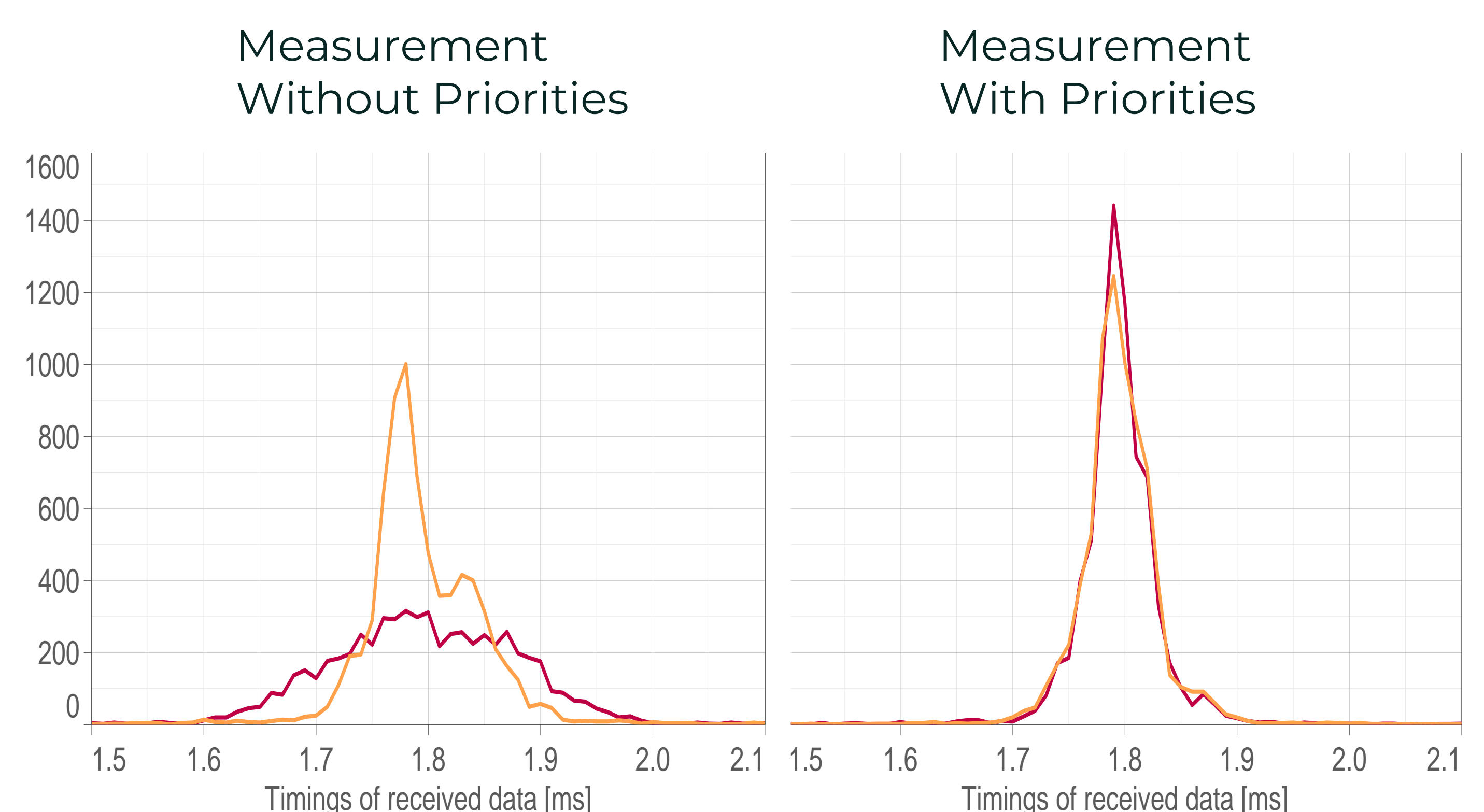
Our methodology involves a comprehensive workflow that begins with architectural design and progresses through Software-in-the-Loop (SiL) and Hardware-in-the-Loop (HiL) validation. We limit DDS topics to a Credit-Based Shaper VLAN connection, which allows us to manage data flow effectively while maintaining the necessary timing constraints. This approach ensures thorough testing and validation of the system's performance under various conditions.

Analysis Results

The results of our analysis demonstrate significant improvements in development efficiency. The integration of DDS and TSN facilitate the usage and generation of artifacts from the architecture, streamlining both the development process and the hardware setup. We achieve time synchronization with a precision of less than 300 nanoseconds, and we expect that our DDS traffic is robust against lower priority data traffic. Our solution not only enhances data distribution flexibility but also maintains the determinism needed for critical automotive functions, validating the success of our integrated approach.

Discussion and Conclusion

In conclusion, our project demonstrated the feasibility and benefits of integrating DDS with TSN in a highly synchronized zonal architecture for automotive applications. The results highlighted significant improvements in development efficiency and communication reliability. The precise time synchronization and robustness of DDS traffic against lower priority data traffic underscore the potential of this approach to meet the stringent requirements of modern automotive systems. Future work will focus on further optimizing the integration and exploring additional use cases within the automotive domain.



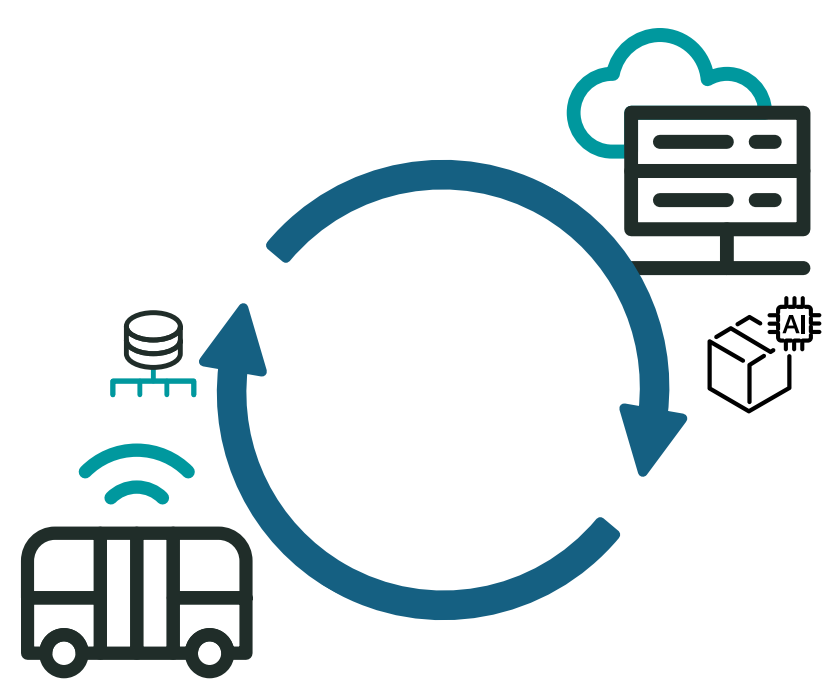
Contributing Partners



Integration and Safety Assurance Methodologies

Digital Loop

Modern service-oriented architectures support upgradable functions in connected vehicle systems. Especially AI-based services can be updated with re-trained models using data from unseen scenarios. This data is continuously collected using an intelligent data recording approach, where recordings are triggered by suitable monitors observing in- and outputs of the AI-model.



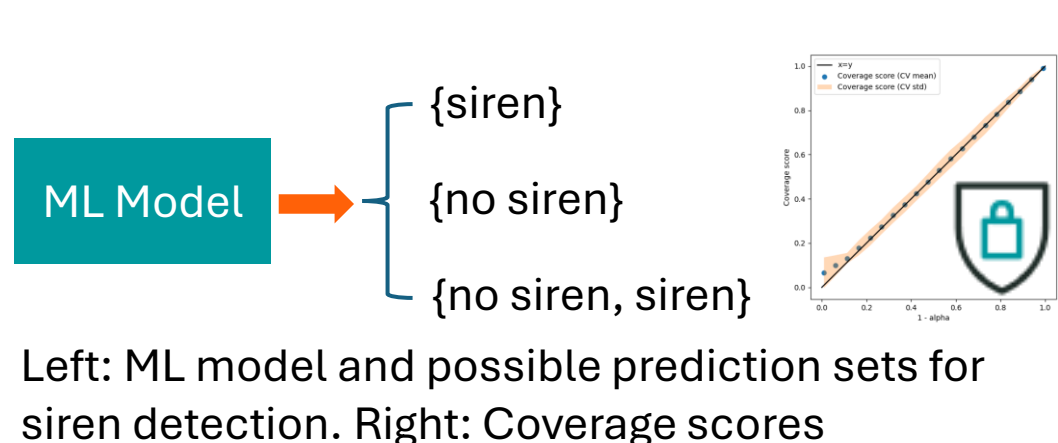
1. Monitor model and input data
2. Trigger data recording based on monitoring events
3. Upload recorded data to cloud backend to improve ML components and their testing

Acoustic Perception

To support a positive driving experience, the shuttle recognizes siren sounds and calls for help. Detection is based on the acoustic analysis of noise or voice signals in the shuttle and its surrounding. Here, we have worked out the Digital Loop for the AI-based siren-detection function.

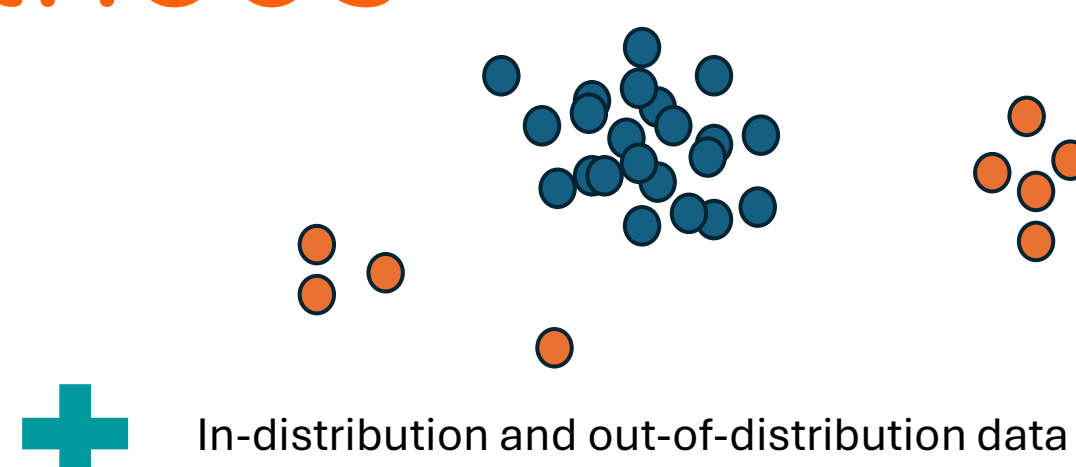


Monitoring Methods



Left: ML model and possible prediction sets for siren detection. Right: Coverage scores
I guarantee that 95% of my prediction sets contain the ground truth for known input data...

Conformal prediction (CP)



I can check if the input data is unknown...

Out of distribution (OOD) detection

In-Distribution-Data: Conformal Prediction

For monitoring and service-self-assessment, we use the model's uncertainty estimates. This requires that they are reliable. Confidence scores from ML-models are usually not well calibrated and thus cannot be trusted for safe decision making. Conformal prediction addresses this problem and provides provable guarantees that the true prediction is covered by the prediction set with a prescribed probability. This only holds for data from the same distribution as the calibration data. Unfortunately, we can not rule out that the deployed model faces unknown (OOD) noises, which violates this assumption.

Monitoring Methods

Out-of-Distribution (OOD) Data: OOD Detection

We choose a suitable out-of-distribution (OOD) detection method which is based on the embeddings of the underlying model. It helps to identify OOD data from new and unseen scenarios. This serves as a trigger for data recordings. In the service-self-assessment it addresses the problem of unknown-unknowns described in the SOTIF standard.

Data Driven Test Methods

Data augmentation: Upsample rare data using mix augmentation

Error analysis using automatic annotations: Did you consider bagpipes in your training data or operational design domain specification?

SNR Robustness: False-negative-rate depending on SNR-level

Explainable AI: Which part of the input data has triggered the decision for "siren"

Once recorded data is uploaded to the cloud, it can be used to enrich training- and test-data pools. Furthermore, the amount of data can be increased by augmentation methods. For the siren-detection use case, we can test robustness of models with respect to background noises. To this end, we automatically annotate large datasets with 521 audio classes and augment our labeled test data with these noises at varying signal-to-noise-ratio (SNR) levels. This allows to spot model weaknesses for certain background noises or SNR levels. This information can then be used to adapt training data sets or the operational-design-domain specification for the function. To analyze single predictions, we employ explainable AI methods such as guided grad cam.

Contributing Partners



Integration and Safety Assurance Methodologies

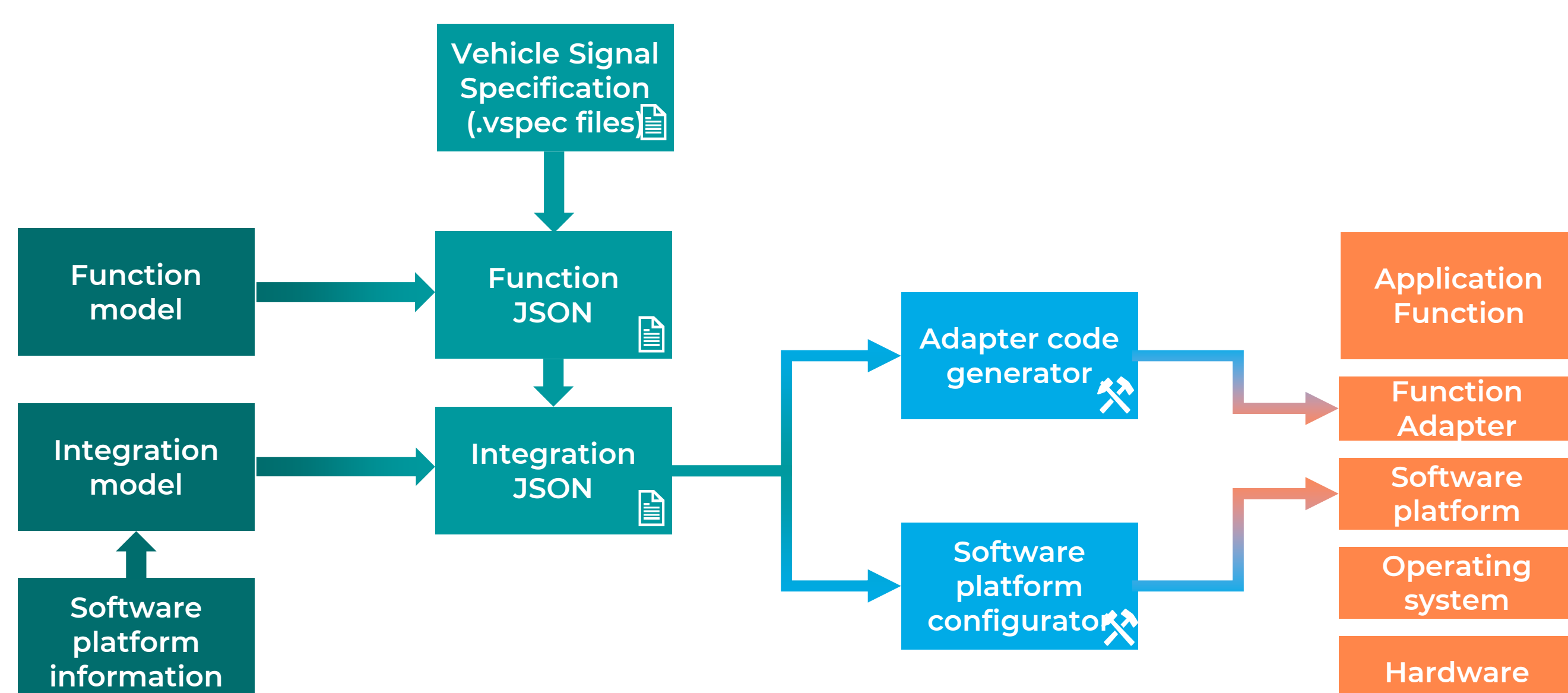


Goal of the Project

One of the project's objectives is to develop methods and tools that facilitate the seamless integration of application functions across various service-oriented software architecture platforms, including Robot Operating System 2 (ROS 2) and AUTOSAR Adaptive Platform (AUTOSAR AP).

Our Methods to reach this goal

- To enable efficient integration onto multiple software platforms, it is necessary to abstract the necessary information regarding application and software platforms and describe in a standardized manner.
- A **Function Model** using an object model is created to capture all the relevant information of the application such as data, parameters, errors and so on. Here, the data and parameter interfaces are standardized based on Vehicle Signal Specification (VSS).
- An **Integration Model**, using an object model, is developed as a standardized model to capture the integration specifications of an application with respect to a software platform. This includes service definition, configuration of service protocols (SOME/IP, DDS), persistency, diagnostics and more. The Integration Model is derived from the Function Model and the corresponding software platform.
- The **Integration JSON** generated from the Integration Model is then consumed by **tools** to configure the software platform automatically and to generate adapter code. This adapter code maps the application to the corresponding APIs of the chosen software platform.



Our Proof of Concept

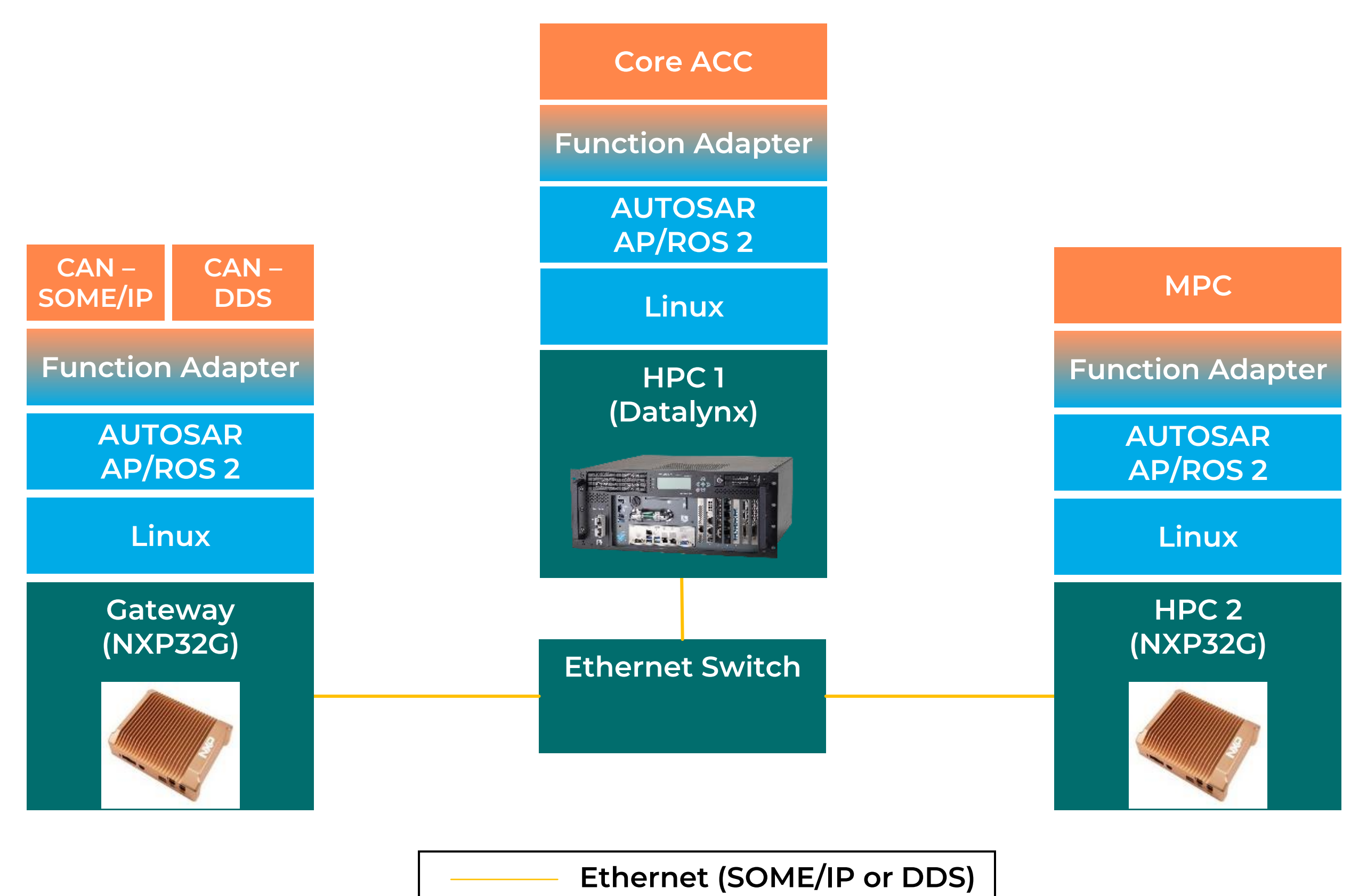
To demonstrate the proposed methodology, the workflow has been applied to integrate an Advanced Driver Assistance System (ADAS) application namely **EcoControl onto ROS 2 and AUTOSAR AP**. The data interfaces of the application are standardized based on VSS. Figure below shows the system architecture of the application.

ROS 2:

- Integration JSON and adapter code are manually coded.
- Application is successfully integrated onto target hardware (Datalynx and NXP32G).

AUTOSAR AP:

- Integration JSON is manually generated.
- Application Framework** (tool developed in collaboration with Vector Informatik GmbH) is used for auto configuration of AUTOSAR AP in DaVinci Developer Adaptive and generation of adapter code.
- The application is successfully integrated onto AUTOSAR AP.



Contributing Partners

