ReliablE in-Vehicle pErception and decisioN-making in complex environmenTal conditionS

Grant Agreement Number: 101069614

D.2.2 Full Stack Architecture & Interfaces

Document Identification			
Status	Final	Due Date	31/06/2023
Version	1.0	Submission Date	28/06/2023
Related WP	WP2	Document Reference	D.2.2
Related Deliverable(s)	D2.1	Dissemination Level	PU
Lead Participant	HIT-FR	Document Type:	R
Contributors	All WP2 partners	Lead Authors	Anthony Ohazulike, HIT-FR
		Reviewers	Thomas Griebel (UULM)
			David Fernandez (APTIV) Kirsty Aquilina (APTIV)

Funded by the European Union

This project has received funding under grant agreement No 101069614. It is 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.

Author(s)		
First Name	**Last Name**	**Partner**
Seshan	Venkita	APTIV
Fabio	Tango	CRF
Anthony	Ohazulike	HIT-FR
Quan	Nguyen	HIT-FR
Alireza	Ahrabian	HIT-UK
Bill	Roungas	ICCS
Anastasia	Bolovinou	ICCS
Markos	Antonopoulos	ICCS
Henri	Terho	S4
Mohit	Mehndiratta	S4
Leonardo	Gonzalez	TECN
Asier	Arizala	TECN
Dariu	Gavrila	TUD
Ted	de Vries Lentsch	TUD
Michael	Buchholz	UULM
Konstantinos	Koufos	WMG
Mehrdad	Dianati	WMG

Document History			
Version	**Date**	**Modified by**	**Modification reason**
0.1	12/05/2023	HIT	Initial version (structure and some preliminary content)
0.2	22/05/2023	All partners	Comments and input from all partners
0.3	01/06/2023	HIT	Updated version based on partners' comments and final revision before is sent to the reviewers
0.4	16/06/2023	UULM	Document was reviewed by UULM
0.5	19/06/2023	APTIV	Document was reviewed by APTIV
0.6	21/06/2023	HIT	Reviewers' comments/suggestions addressed by HIT
0.7	26/06/2023	ICCS	Final review by the project's coordinator
1.0	28/06/2023	ICCS	Final version

Quality Control		
Role	**Who (Partner's short name)**	**Approval Date**
Deliverable leader	Anthony Ohazulike (HIT-FR)	27/06/2023
Quality manager	Panagiotis Lytrivis (ICCS)	27/06/2023
Project Coordinator	Angelos Amditis (ICCS)	28/06/2023

Executive Summary

The "D2.2: Full Stack Architecture & Interfaces" deliverable is a public report of the EVENTS project, which main goal is to provide both high-level and detailed system architecture and interfaces of the Experiments (EXPs) that will be developed in EVENTS. These EXPs, which have a basis on the three Use Cases (UCs) defined in the Grant Agreement (GA) have been described in Deliverable D2.1. This deliverable is tied to WP2/Task 2.3 which aims to identify functional clusters and interfaces so that allocation and decomposition of vehicle and system-level requirements into components is achieved regarding EXPs defined in WP2/T2.1.

To enhance the technical collaboration within the consortium and unify the autonomous driving stack that will be developed as part of EVENTS, a high-level general architecture which encapsulates all the eight EVENTS' EXPs was designed and adopted by all members. Following this high-level architecture and nomenclature, a high-level and a detailed architecture and interfaces were designed and described for each of the eight EXPs.

Having a basis in T2.1 and T2.2, activities of T2.3 are documented in this deliverable. The detailed architectures and interfaces described in this deliverable form the basis of the software (SW) stack and technical development for EVENTS.

Table of Contents

List of Tables

List of Figures

Abbreviations & Acronyms

Abbreviation / acronym	Description
ACC	Adaptive Cruise Control
AD(F)	Autonomous Driving (Function)
AI	Artificial Intelligence
AL	Alert Limit
AV	Automated Vehicle
BP	Behavioural Planner
CA	Consortium Agreement
CAM	Cooperative Awareness Message
CAV	Connected Automated Vehicle
CPM	Collective Perception Messages
DDT	Dynamic Driving Task
DENM	Decentralized Environmental Notification Message
DM	Decision Making
EC	European Commission
EXPs	Experiments

Abbreviation / acronym	Description
FIS	Fuzzy Inference System
FoV	Field of View
FTP	Fail-degraded Trajectory Planning
GA	Grant Agreement
IR	Integrity Risk
ISO	International Organization for Standardization
I/O	Input(s) / Output(s)
LiDAR	Light Detection and Ranging
MDP	Markov Decision Process
MPC	Model Predictive Control
MRM	Minimum Risk Manoeuvre
MOP	Moving Object Prediction
MOT	Multi-Object Tracking
ODD	Operational Design Domain
PE	Position Error
PL	Protection Level
PP	Perception Platform
RADAR	RAdio Detecting And Ranging
REQs	Requirements
RL	Reinforcement Learning
SAE	Society of Automotive Engineers
SMD	Safety-mode Decision
SPaT message	Signal Phase and Timing message
SPECs	Specifications
TOR	Take Over Request
TP	Trajectory Planner
TSs	Target Scenarios
UCs	Use Cases
VRU	Vulnerable Road User
WP	Work Package

1. Introduction

Following the Use Cases (UCs) and experiments (EXPs) described in Deliverable D2.1: "User and System Requirements for selected Use-cases" (which will be summarized in this current document), Deliverable D2.2 "Full Stack Architecture & Interfaces" presents the overall system and Software (SW) architecture for all the 8 EXPs as described in Section 4 of this document. First, a high-level architecture (including terminologies) for the entire set of EVENTS' EXPs (referred to as "the Master Architecture") was designed and adopted by all EXP Partners. Then, based on the adopted Master Architecture and terminologies, a subset high-level architecture was created for each EXP. Finally, based on this subset architecture, a detailed architecture was created for each EXP, and this current deliverable gives details of these architecture for all EXPs. These architectures form a base guide for SW development for all the EXPs.

1.1 Aim of the Project

Driving is a challenging task. In our everyday life as drivers, we face unexpected situations we need to handle in a safe and efficient way. The same is valid for Connected and Automated Vehicles (CAVs), which also need to handle these situations, to a certain extent, depending on their automation level. The higher the automation level is, the higher the expectations for the system to cope with these situations are.

In the context of this project, these unexpected situations where the normal operation of the CAV is close to be disrupted (e.g., ODD limit is reached due to traffic changes, harsh weather/light conditions, imperfect data, sensor/communication failures, etc.), are called "events".

Today, CAVs are facing several challenges (e.g., perception in complex urban environments, Vulnerable Road Users (VRUs) detection, perception in adverse weather and low visibility conditions) that should be overcome to be able to handle these events in a safe and reliable way.

Within our scope, and to cover a wide area of scenarios, these "special" events are clustered under three main UCs: a) Interaction with VRUs, b) Non-Standard and Unstructured Road Conditions, and c) Low Visibility and Adverse Weather Conditions.

Our vision in EVENTS is to create a robust and self-resilient perception and decision-making system for AVs to manage different kinds of events on the horizon. These events result in reaching the CAV ODD limitations due to the dynamically changing road environment (VRUs, obstacles) and/or due to imperfect data (e.g., sensor and communication failures). The CAVs should be able to operate safely in these challenging conditions. When the system cannot handle the situation, an improved minimum risk maneuver should be put in place.

1.2 Purpose of the Document

This deliverable (D2.2) is part of the work in Work Package 2 (WP2): "Use cases, requirements and system design" and its main objective is to illustrate in detail the sensor suite and modules that are required to achieve each of the 8 EXPs. It details end-to-end sensor and SW modules (and their inter-connections) needed to realize each of the EXPs described in Section 4. In other words, D2.2 delivers a holistic system view of each EXP.

Described in this document is the base for all developments that will occur in EVENTS, and this document would serve as a regular reference to one who intends to have a glimpse of the working principles behind each of the EXPs. In fact, all developments in WP3 ("Perception and self-assessment") and WP4 ("On-board decision-making for fail-safe automated vehicle motion"), and most development in WP5 ("System integration and safety compliance"), will be based on details described in this document (D2.2).

Though this document is a direct consequence of Deliverable D2.1 "User and System Requirements for selected Use-cases", the effort has been made to make this current document (D2.2) a stand-alone document, hence all necessary information needed to understand this document will be summarized herein.

1.3 Related EVENTS Tasks and Terminology

There are two main terms to clearly define, Use Cases (UCs) and Experiments (EXPs). UCs are the abstract use cases that were initially defined in the Grant Agreement (GA) and are summarized in this document (Section 2).

More specifically, a UC is a collection of related aspects of the operational design in which the system will be deployed, along with the desired behaviour of the AV. Our UCs are described from the AV perspective, meaning that the AV needs to react to a certain traffic situation/condition.

Experiments (EXPs) are the specific realizations of one or more UCs by each partner or by synergies of more than one partner. EVENTS defined 8 EXPs and they are summarized in Sections 2.2 & 3.

An overview of the structure of WP2, which demonstrates the methodological workflow from the UCs to the EXPs definition (Task T2.1), to the REQs collection and analysis (Task T2.2), to the EVENTS systems' and sub-systems' architecture (Task T2.3) and finally, to the hazard analysis & risk assessment (Task T2.4) is shown in Figure 1.

1.4 Intended Readership

This deliverable (D2.2) is the backbone of the EVENTS project and will serve as a handy document for readers interested in Autonomous Driving, and most especially readers who would like to have a bit detailed insight of how CAVs are designed to fulfill the objectives of the EVENTS's EXPs.

As stated before, this document would serve as a reference/guide for all EXP participants and all consortium members. From this document, members can identify modules of interest from other EXPs and seek more information or cross-collaboration from the owners of that module. Further, this document will serve as a handbook for all WP3, WP4, and WP5 participants.

1.5 Document Overview

This document starts with an introduction on the EVENTS project, the purpose of the current document, and the structure of WP2. Section 2 briefly summarizes the UCs of EVENTS, followed by a high-level summary of 8 EVENTS' EXPs. Section 3 describes the overall EVENTS' high-level architecture (otherwise known as the Master Architecture) and agreed terminologies. Section 4 gives more information on each of the 8 EXPs and detailed the full stack architecture & interfaces[1] for each of the EXPs. Finally, Section 5 concludes the document.

[1] Full Stack Architecture & Interfaces refers to both high-level and detailed system architecture and interfaces of the Experiments (EXPs), showing main modules that will be developed and deployed.

2. Brief Use Cases and Experiment Description

2.1 Use Cases

2.1.1 UC1: Interaction with Vehicles and VRUs in Complex Urban Environment

UC1 is concerned with safe and resilient AD in complex urban environments, i.e., cluttered surroundings (occlusions), multiple road users, or V2X-assisted intersections. Focus is given on interacting with Vulnerable Road Users (VRUs), e.g., pedestrians, or cyclists. Urban roundabouts are also of interest in this UC with a focus on V2V integration and advanced control based on collective perception. The participating partners are (in alphabetical order): 1. ICCS, 2. TECN, 3.TUD, and 4. UULM.

2.1.2 UC2: Non-Standard and Unstructured Road Conditions

UC2 investigates non-standard and unstructured road conditions, for example, road works, accident zones, or park areas with no lane markings. The participating partners are (in alphabetical order): 1. APTIV, 2. CRF, 3. HIT-FR, 4. HIT-UK, 5. TECN, and 6. WMG.

2.1.3 UC3: Low Visibility & Adverse Weather

UC3 aims to extend the environmental conditions of AD functions. The participating partners are (in alphabetical order): 1. APTIV, 2. HIT-FR, 3. HIT-UK, 4. ICCS, 5. S4, and 6. WMG.

2.2 Brief Description of the Experiments

The following 8 EXPs stemming from the 3 UCs in Section 2.1 will be realized in EVENTS.

EXP1: Interaction with VRUs in complex urban environment.

EXP2: Re-establish platoon formation after splitting due to roundabout.

EXP3: Self-assessment and reliability of perception data with complementary V2X data in complex urban environments.

EXP4: Decision making for motion planning when faced with roadworks, unmarked lanes, and narrow roads with assistance from perception self-assessment.

EXP5: Decision making for motion planning when entering a jammed highway.

EXP6: Small object detection at a far range in adverse weather conditions.

EXP7: Localization/perception self-assessment for advanced ACC and other vehicles' behaviour prediction under adverse weather or adverse road conditions.

EXP8: Driving minor road under adverse weather conditions including perception self-assessment.

Table 1 below summarizes each partner's contributions to the EXPs.

Table 1: Contributions of each partner in the different EXPs

Experiment	Partners' Contributions
EXP1	**TUD:** Full AD stack supporting urban driving focusing on VRUs interaction
EXP2	**TECN**: Vehicle, motion control, V2V communication **ICCS**: Self-assessment on simulation collaborative driving for CAMs and CPMs
EXP3	**UULM**: Self-assessment of perception, V2X, vehicle infrastructure pilot site or respective data
EXP4	**CRF**: Decision making, vehicle, motion control **HIT**: Perception platform and localization **TECN**: Real-time motion control **WMG**: Perception self-assessment
EXP5	**CRF**: Decision making, vehicle, motion control **HIT**: Perception platform and localization **TECN**: Real-time motion control
EXP6	**APTIV**: Sensing of small objects and semantic representation of the environment (object relative position, object lane assignment, object size, object velocity, object over-drivability and estimation of time to collision).
EXP7	**WMG**: Integrity monitoring and perception self-assessment **ICCS**: Behavior prediction of other vehicles
EXP8	**S4**: Visibility assessment of the lidar, sensor data quality assessment, path planning around a static object on the path, decision making for remote operator assistance, Minimum Risk Manoeuvre decision making.

For the interested reader who would like to know more about these UCs, especially with regards to perception and decision-making challenges associated with the UCs, and proposed approach for AD systems, and full description of all the EXPs, are referred to Deliverable D2.1 "User and System Requirements for selected Use-cases" [1], while an overview of the relationship between the UCs, EXPs and the tasks of WP2 is shown in Figure 1.

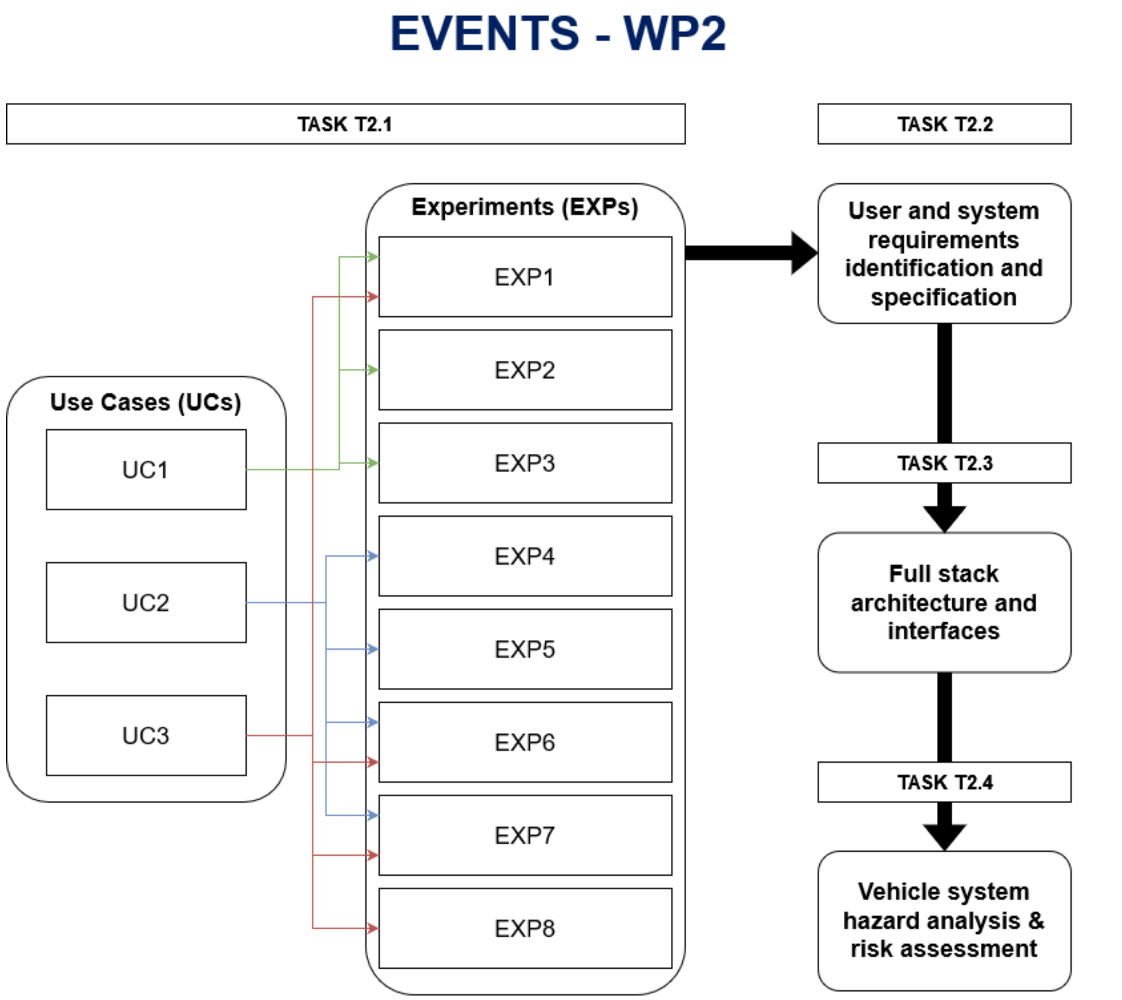


Figure 1: EVENTS WP2 structure [— Related to UC1, — Related to UC2, — Related to UC3]

3. High-level EVENTS Architecture and Interfaces

In this chapter, the high-level architecture of the EVENTS project is illustrated, which is directly linked with the requirements for each experiment, as those are defined in Deliverable D2.1 “User and System Requirements for selected Use-cases” [1].

3.1 Overall EVENTS Architecture and Interfaces

Figure 2 shows the overall high-level EVENTS Full Stack Architecture and Interfaces, together with common terminologies agreed upon by the partners. It shows, at a very high-level, the necessary modules that are needed (from end-to-end) to realize the 8 EXPs of EVENTS. As the Master Architecture, all other EXP-based architectures stem from this base architecture, retaining common terminologies and structure to enable easy identification and mapping of the EXP-based architectures into the Master Architecture.

We use the terms “block” and “module” interchangeably in the rest of this document.

The first block from the left of Figure 2 is divided into 3 main sub-blocks, namely:

- Maps
- Sensors
- V2X Communication

These sub-blocks are envisaged as the data input blocks for Perception and Self-Assessment (SA).

The **Map** sub-module comprises of HD Map, which would provide information such as Lane markings, Traffic landmarks, Traffic light locations, junction information, Drivable areas, etc.

The **Sensors** sub-block provides the list of sensors that will be used in EXPs. Note that each EXP does not necessarily use all the input sub-blocks or sensors, however, all the items listed in the input block have been envisaged to be used in one or more experiments.

The **V2X Communication** sub-block comprises of On-Board-Unit (OBU) and Smart Road-Side Unit (RSU) that will be used to communicate information between vehicles and infrastructure.

The Perception block which mainly receives data from the input block (left-most block), comprises several sub-modules. These submodules are described below:

- **Localization**: This module localizes the ego-vehicle in world or surrounding coordinate and can be used to relatively localize other objects in the scene.

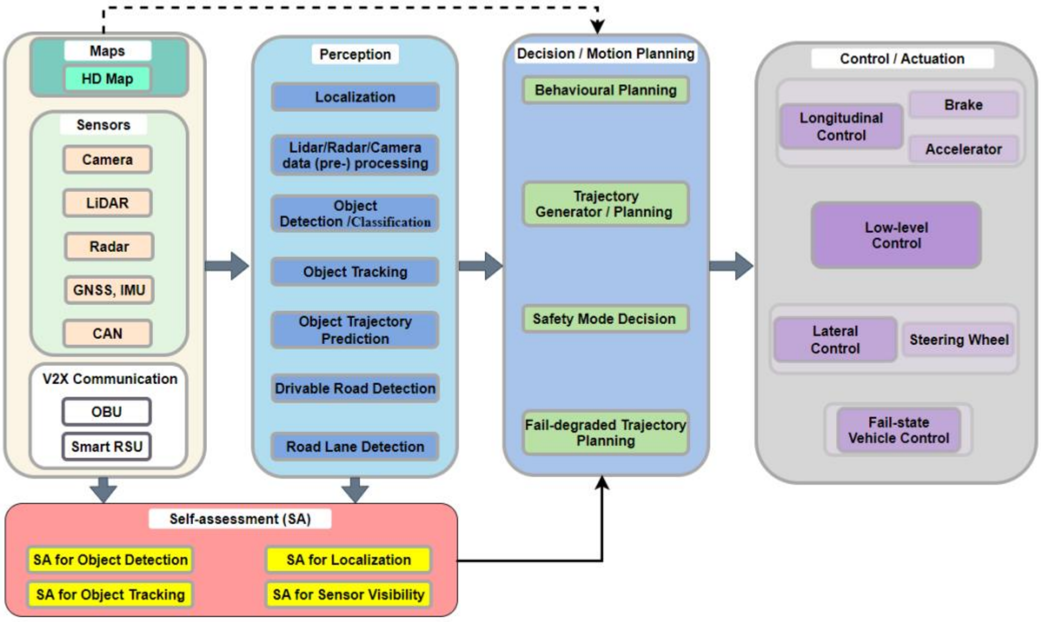


Figure 2: EVENTS high-level Full Stack Architecture and Interfaces ("Master Architecture")

- **Lidar/Radar/Camera data (pre-) processing**: This module processes raw Lidar, Radar, and Camera information, getting it ready to be used by other modules.
- **Object Detection / Classification**: This module detects objects in the scene and classifies what type of objects they are. The module is also responsible for estimating object properties such as relative position and motion state.
- **Object Tracking**: This module estimates (current) detected objects properties based on previous states, aiming at associating the same ID to the same object over time.
- **Object Trajectory Prediction**: The aim of this module is to predict future trajectory, and hence the future position of detected and tracked objects.
- **Drivable Road Detection**: This module is responsible for detecting whether a road is drivable or not.

- **Road Lane Detection**: Lane markings on the roads are detected by this module, and this information is then conveyed to modules or components needing it.

Lying below the Input and Perception blocks is the SA block. This block's main objective is to provide an integrity monitoring mechanism for sensor and perception outputs. The SA block has the following submodules:

- **SA for Object Detection**: This module performs the SA on the output of the perception's object detection module.
- **SA for Localization**: This SA module performs the self-assessment on the output of the perception's localization module.
- **SA for Object tracking**: This module performs the self-assessment on the output of the Object Tracking module of the Perception platform. Note that this module can be an integral part of the Object Tracking module.
- **SA for Sensor Visibility**: As sensor performance might degrade due to many factors, such as dust particles, rain, and fog, this SA module monitors and performs the self-assessment on the output of the sensors.

The next main block is the Decision / Motion Planning block which comprises the following submodules:

- **Behavioral Planning**: This module generally delivers high-level decision-making of driving behaviors such as lane changes, passing of parked cars, and progress through intersections.
- **Trajectory Generator / Planning**: Taking the high-level decision output of the Behavioral Planner as the main input, this module generates a feasible and safe trajectory in form of a series of waypoints, which feeds to the low-level control of the vehicle.
- **Safe Mode Decision**: This module is responsible for selecting the most suitable operation mode for the vehicle according to the evaluation of the vehicle status, as well as the surrounding environment.
- **Fail-degraded Trajectory Planning**: This module provides a minimum risk maneuver (MRM), such as an automated emergency action or a safe lane change (LC) in the emergency lane. Since the MRM is strictly related to the demo vehicle, this module will be developed specifically

by every single demo owner, and thus the manoeuvre will be executed by that vehicle.

The next main block is the Control / Actuation block which comprises several low-level vehicle control components. The **low-level control** architecture or design is specific to each demo vehicle, however, we will give a description that is as general as possible:

- **Longitudinal Control**: This module is responsible for longitudinal control of the ego vehicle. Such control includes speed and acceleration using the **brake** and **accelerator** functions.
- **Lateral Control**: This module is responsible for controlling the ego vehicle in a lateral direction. Such control is usually done by altering the steering wheel angles/speed using the **steering wheel** functions.
- **Fail-state Vehicle Control**: The intended objective for this module is to serve as a monitor for low-level control, assessing all inputs to the controller, as well as in the case of triggering a degraded mode. It is responsible for actuating the MRM.

4. High-level and Detailed Architecture and Interfaces for all EVENTS Experiments

4.1 Experiment 1

4.1.1 Description

Experiment 1 (EXP1) "Interaction with VRUs in complex urban environment" is about safe, comfortable and time-efficient automated driving in complex urban environment while interacting with VRUs (e.g. pedestrians, cyclists). The environment perception, road user motion prediction, motion planning and vehicle control will be demonstrated in a single integrated system on-board a vehicle prototype. The experiment consists of the ego-vehicle driving on a two-lane road (i.e., one lane on each side) whereas several VRUs might (or might not) move into the vehicle's path (e.g., crossing, walk longitudinally, swerve), possibly from behind occlusions (e.g., parked vehicles) – Figure 3. The question is whether to decelerate, accelerate or steer away. Experimentally, some VRUs will be real, while others will be realistic dummies (e.g., 4activeSystems). This experiment will include both benevolent and more challenging environmental/lighting conditions (e.g., night, rain, blinding sun), where visibility is hampered.

TUD is the leader and sole participant of this experiment.

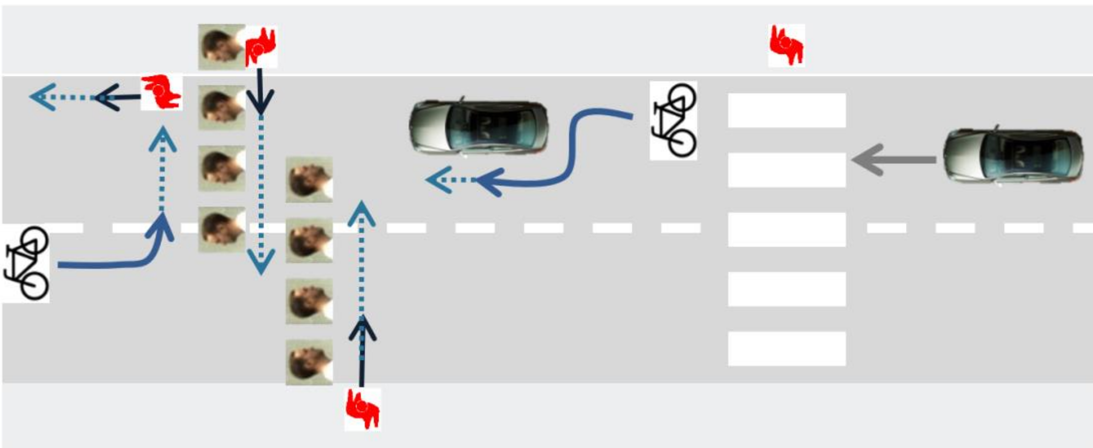


Figure 3: EXP1 Graphical Representation

4.1.2 High-level Architecture and Interfaces

Figure 4 depicts the high-level architecture of EXP1. It is a direct subset of the Master Architecture (Figure 2). Description of the modules are as described in Section 0.

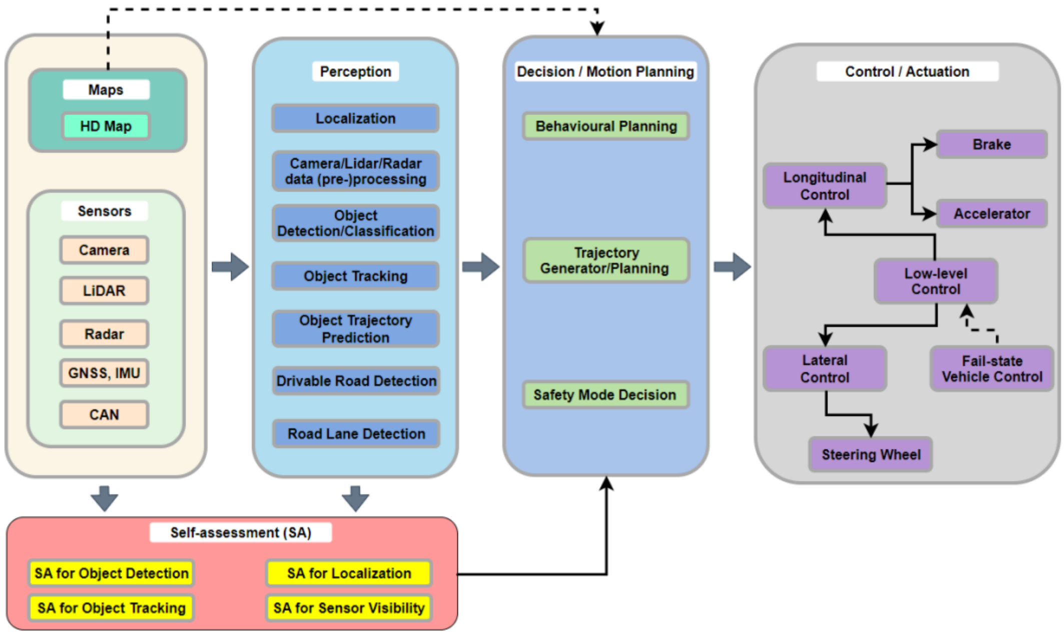


Figure 4: EXP1 high-level Full Stack Architecture and Interfaces

4.1.3 Detailed Architecture and Interfaces

Figure 5a & Figure 5b show the detailed architecture and inter-relation between modules for the realization of EXP1. The main items in the architecture are explained as follows:

Perception Block (Figure 5a)

- **Lidar/Radar/Camera data (pre-)processing**: This module processes the raw **Camera**, **LiDAR**, and **Radar** data. This block outputs the stereo point cloud, and the filtered LiDAR and Radar point clouds. This output is named the "Preprocessed Sensor Data".

- **Object Detection/Classification**: This module uses the "Preprocessed Sensor Data" to detect objects in 3D.

- **Object Tracking**: This module uses the output of Object Detection/Classification in combination with the global vehicle pose (output of "**Localization**" block) to track the objects.

- **Object Trajectory Prediction**: Finally, object trajectories are predicted in this module using the HD map and the object trajectories (output of "Object Tracking" module).

- **Drivable Road Detection**: The drivable road is determined in this module using the preprocessed sensor data (output of the "Lidar/Radar/Camera data (pre-)processing" block), 3D detections (output of the "Object Detection/Classification" block), global vehicle pose (output of "Localization" block), and the HD map.

- **Road Lane Detection**: The road lane coordinates are computed using the preprocessed sensor data (output of the "Lidar/Radar/Camera data (pre-)processing" block), the global vehicle pose (output of "Localization" block), and the HD map in the "**Road Lane Detection**" module.

- **Localization**: The global vehicle pose is determined in this module using the latitude and longitude (GNSS output), the vehicle velocities and accelerations (IMU output), the preprocessed sensor data (output of the "Lidar/Radar/Camera data (pre-)processing" block), and the HD map.

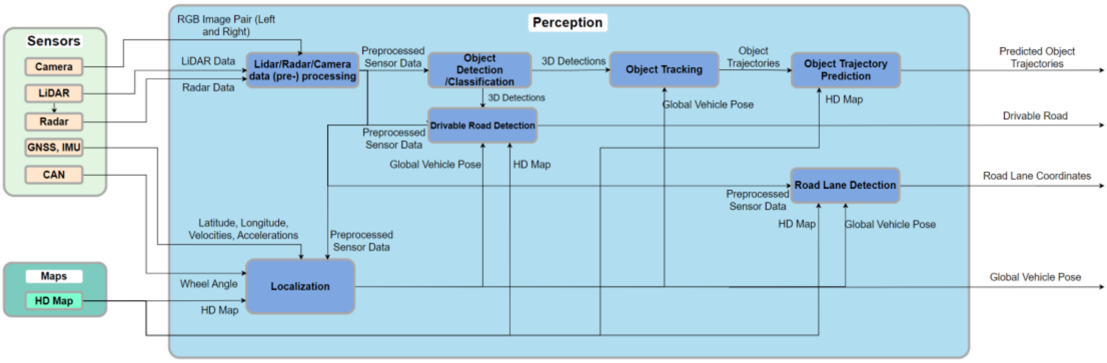


Figure 5a: EXP1 Detailed Full Stack Architecture and Interfaces (Perception)

<u>Decision / Motion Planning Block (Figure 5b)</u>

- **Perception Block Outputs**: is a pseudo block to easy the visualization of the "Decision / Motion Planning" stack, and it contains the outputs of the "Perception" stack described in Figure 5b.
- **Behavioural Planning**: The global vehicle pose (output of "Localization" block), and the HD map are used by this module to determine the behavior of the ego-vehicle.
- **Trajectory Generator/Planning**: This module uses the global vehicle pose (output of "Localization" block), waypoint (output of "Behavioural Planning" block), predicted object trajectories (output of "Object Trajectory Prediction" block), road lane coordinates (output of "Road Lane Detection" block), drivable road (output of "Drivable Road Detection" block), and the HD map to determine the vehicle target trajectory. This trajectory is the input of the "**Control/Actuation**" block.
- **Safety Mode Decision**: This module uses the self-assessment score to determine whether the safety driver should take over the control of the AV. In case the safety driver should take control, a warning will be shown.

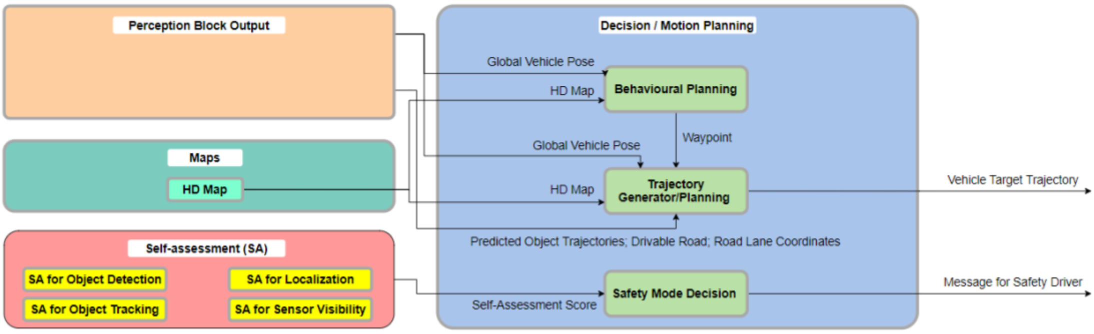


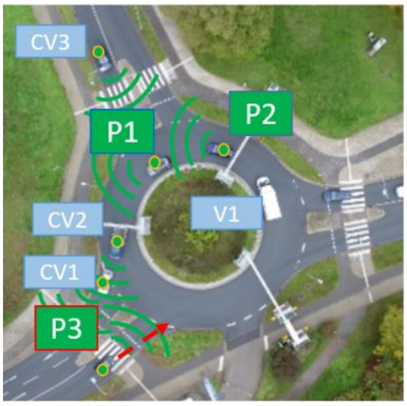
Figure 5b: EXP1 Detailed Full Stack Architecture and Interfaces (of Decision / Motion Planning)

4.2 Experiment 2

4.2.1 Description

In Experiment 2 (EXP2) "Re-establish platoon formation after splitting due to roundabout", a coordinated platooning planning is investigated via V2X data integration. A platoon ensemble composed by three CAVs (one CAV leader and two CAVs as followers) in an urban environment is assumed, which is split because of traffic when approaching and crossing a roundabout (driving rules in the roundabout are assumed to prioritize the vehicles inside the roundabout). The followers should be able to reach the leading vehicles ensuring string stability also under curved trajectories. Planning of re-joining the platoon takes advantage of Collective Perception Messages' (CPMs) fused info (and confidence) that is made available to the follower when entering the roundabout (see Figure 6).

TECN is the leader of this Experiment and **ICCS** is a partner.



- Green dot denotes V2X capability of the traffic agent

P1: CAV platoon leader
P2: CAV platoon follower #1
P3: CAV platoon follower #2
(←this is the subject vehicle which tries to reconnect with the platoon via merging into the roundabout right after/before P1, P2, details of P1, P2, P3 choreography so that a platooning reconnection is realized to be specified later)
CV1, CV2, CV3: Connected vehicles able to share CAM, DENM, CPM info
V1: not connected vehicle

Figure 6: EXP2 Graphical Representation[2]

4.2.2 High-level Architecture and Interfaces

Figure 4 depicts the high-level architecture of EXP2. It is a direct subset of the Master Architecture (Figure 2). Description of the modules are as described in Section 0.

[2] DENM: Decentralized Environmental Notification Message

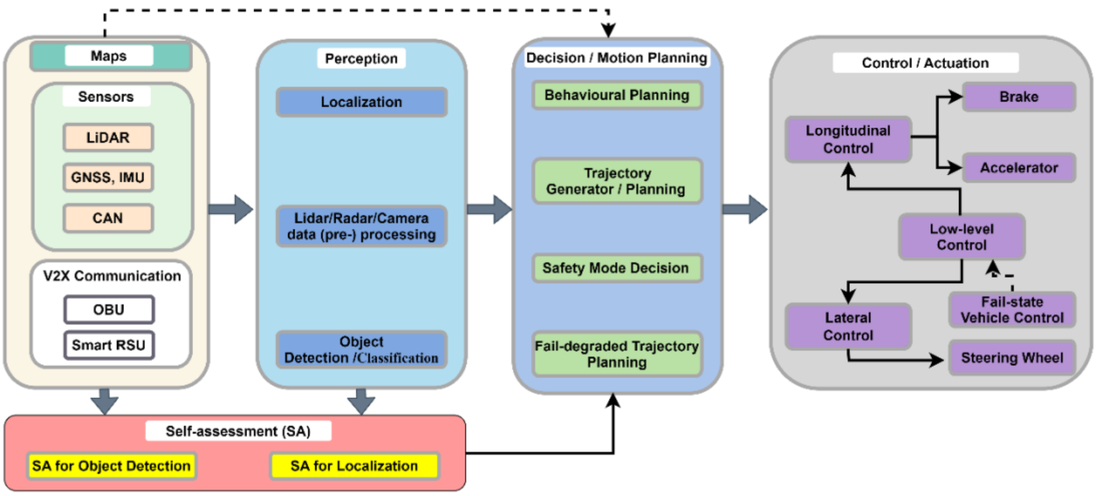


Figure 7: EXP2 high-level Full Stack Architecture and Interfaces

4.2.3 Detailed Architecture and Interfaces

Figure 8 shows the detailed architecture and inter-relation between modules for the realization of EXP2. The main items in the architecture are explained as follows:

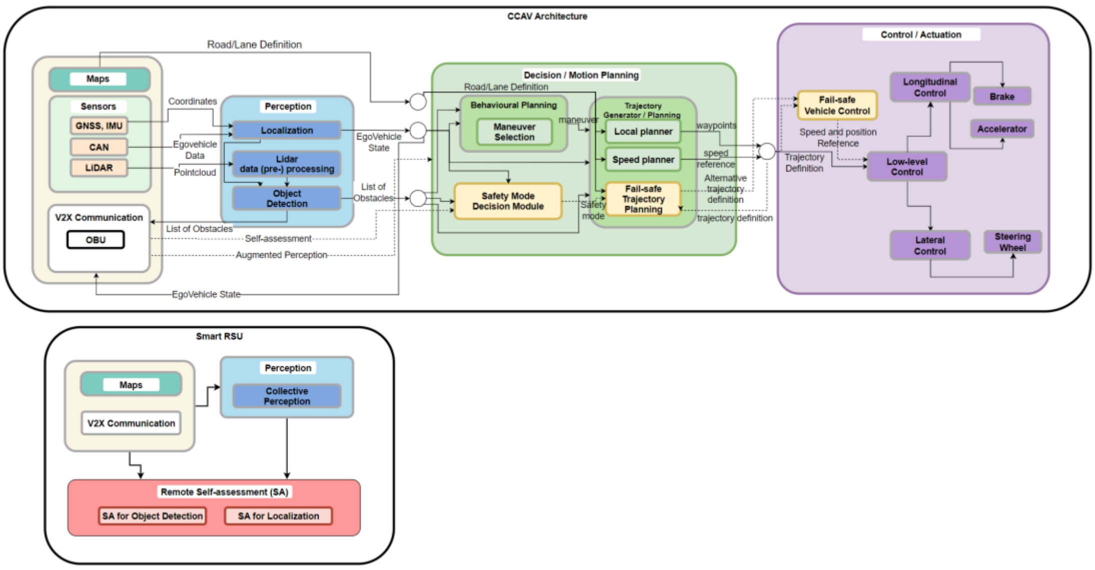


Figure 8: EXP2 Detailed Full Stack Architecture and Interfaces

- **Sensor/Map/Communication/Perception**: The perception module will allow the vehicle platforms in EXP2, the ability to detect surrounding vehicles and localize themselves in their environment. To achieve this a series of sensors are used; a differential GNSS will be available for precise positioning, augmented with an IMU and predictive model. Aiding to localization and detection as well, a Lidar configuration will be installed in each vehicle. Simulated platforms will also have sensing capability which allows our perception stack to be integrated. V2X communication will also enhance the situation understanding by integrating collective perception information.

- **Collective Perception & Self-assessment**: The Collective Perception module receives a map and individual CCAV perception data at the object level (V2I) and fuses them to construct a BEV map of the scene (i.e., the collective perception of the participating CCAVs) in terms of a probabilistic occupancy grid. The probabilistic certainty of the resulting grid plus further consistency/plausibility checks between the resulting collective perception and each individual CCAVs (claimed) Field of View (FoV) and perception data will be utilized for assessing the reliability of the result and potential misbehavior of participating nodes.

Decision/Motion Planning

Decision module will integrate inputs from the perception module, map information, vehicle proprioceptive sensors and V2X information to provide a feasible and safe trajectory, from the maneuver level to a parametric curve definition and speed planning to the low-level control systems.

- **Behavioral Planning**: This software module is a part of the Decision/Motion planning, which will take care of high-level decision making in the form of a maneuver selection considering disturbances and unexpected behaviors from other agents. This module will be developed in the context of platooning and roundabout navigation.
- **Trajectory Generator/Planning**: This software module will be tasked with generating a feasible and safe trajectory in real time, taking as input a high-level decision or maneuver, and giving as output a series of waypoints for the low-level control of the vehicle. This module will be developed in the context of platooning and roundabout navigation. It will be subdivided into three parts; a local planner, which will define the path based on the maneuver and map definition, a speed planner which will define speed profiles to accomplish the maneuvers in a safe and comfortable manner and a fail-safe trajectory which will compute additional trajectories to be used or weighted against the default configuration when in degraded mode, or if necessary due to input from the Safety Mode Decision Module.
- **Safety Mode Decision Module**: This software module is responsible for selecting the most suitable operation mode for the vehicle according to the evaluation of the vehicle status, as well as the surrounding environment. This evaluation is made during the whole operation, and there are three possible modes of operation. They include normal operation, fail-safe operation and total failure-automation disengagement. During normal operation, the vehicle will continue driving as intended. Fail-safe operation

means that the vehicle cannot maintain the intended goal, but appropriate measures can be taken to bring the vehicle into a safe state (which may imply driving for a limited period). Last, if the situation is dangerous enough to continue driving, the vehicle will be stopped immediately, and the automation disengaged.

Control

- **Fail-Safe Vehicle Control**: This module will work as a monitor for low-level control, to assess the inputs given to a controller, or in case a degraded mode is triggered. It will consider robust control algorithms to provide a safe tracking of the reference provided to lower-level control.
- **Low-level Control**: Low-level control is specific for each demo owner, in the case of EXP2, this control interacts with the electronic units in the vehicle to provide appropriate inputs to the lateral and longitudinal actuators, and it considers the vehicle model and actuator characteristics.

4.3 Experiment 3

4.3.1 Description

Experiment 3 (EXP3) "Self-assessment and reliability of perception data with complementary V2X data in complex urban environments" is concerned with safe automated driving in a complex urban environment with occlusion, to demonstrate the integration of reliability assessment outputs of environment state estimation (onboard self-assessment methods) and V2X data into an onboard perception system. The experiment will be conducted both in a virtual and a real environment. The former will be simulation-based, and it will be primarily concerned with developing a self-assessment layer for the perception data (Task 3.5) along with complementary V2X data (Task 3.4). The latter will be realized in a demo vehicle, with safety drivers/marshals to account for the prototypical status of the developed system, and in a V2X infrastructure pilot site, where the automated ego vehicle will face objects and (artificial) error/degradation in one of the sensors/V2X.

UULM is the leader and sole participant in this experiment.

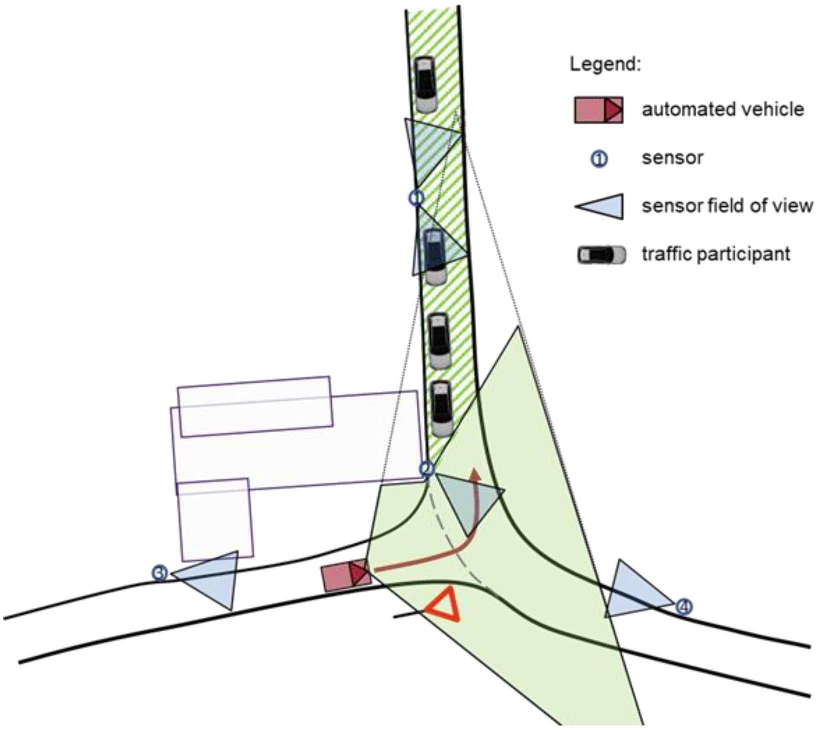


Figure 9: EXP3 Graphical Representation

4.3.2 High-level Architecture and Interfaces

Figure 10 depicts the high-level architecture of EXP3. It is a direct subset of the Master Architecture (Figure 2). Description of some important modules are given below, and one can find the rest as described in Section 0.

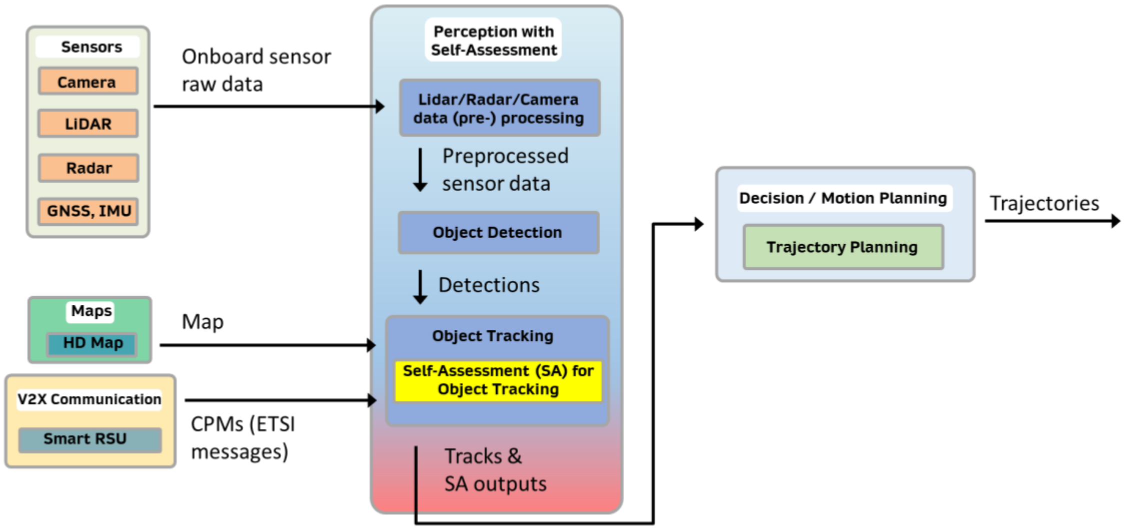


Figure 10: EXP3 high-level Full Stack Architecture and Interfaces

- **Object tracking**: This module tracks static and dynamic objects in the scene, by V2X data, extended field of view (FOV).

  - Input: Detections from onboard sensor data, CPMs from infrastructure pilot site
  - Output: Tracks in the extended FOV

- **SA for object tracking (included in the object tracking module)**: This module self-assesses the tracking performance and its reliability online.
  - Input: Parameters and configurations of tracking, detections from onboard sensor data, CPMs from infrastructure pilot site
  - Output: SA measures for monitoring the tracking performance online

- **Trajectory planning**: This module takes SA scores of tracking into account during trajectory planning (e.g., reduced speed in case of high uncertainty).
  - Input: Tracks, object tracking SA scores
  - Output: Trajectories based on tracking with SA scores

4.3.3 Detailed Architecture and Interfaces

Figure 11 shows the detailed architecture and inter-relation between modules for the realization of EXP3 with a focus on Self-Assessment for Object Tracking. The main items in the architecture are explained as follows:

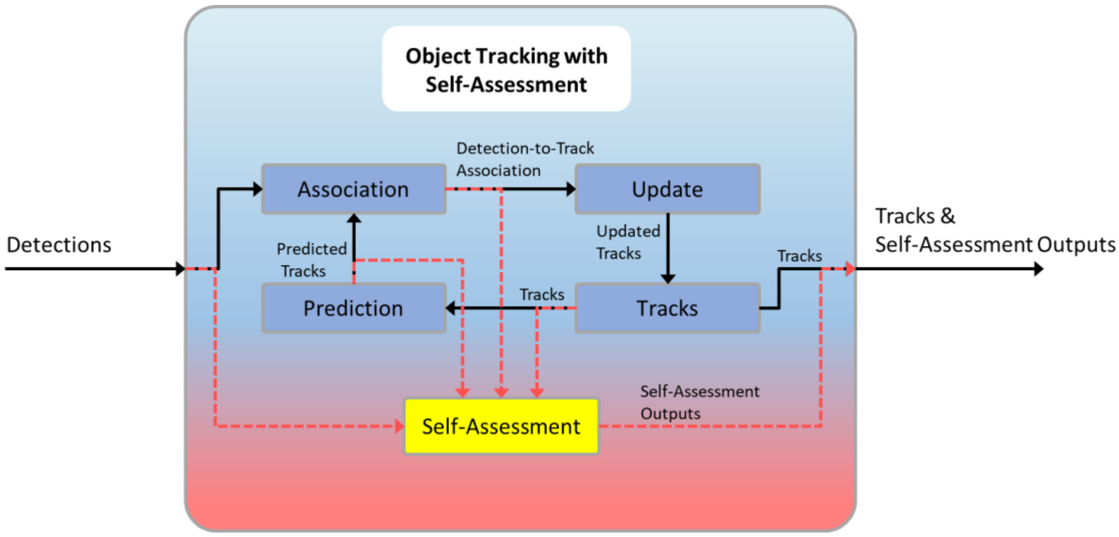


Figure 11: EXP3 Detailed Full Stack Architecture and Interfaces

- **Association**: This module performs detection-to-track association.
  - Input: Detections
  - Output: Detection-to-track association

- **Update**: This module is responsible for updating tracks with the detection-to-track association.
  - Input: Detection-to-track association
  - Output: Updated tracks
- **Prediction**: The main task of this module is predicting tracks to the next time step using the motion model.
  - Input: Tracks
  - Output: Predicted tracks
- **Self-Assessment**: This module self-assesses and monitors the tracking performance and its reliability online
  - Input: Detections; Detection-to-track association; Predicted tracks and updated tracks; Parameters, and configuration of tracking algorithm.
  - Output: Self-assessment measures

4.4 Experiments 4 & 5

4.4.1 Description

4.4.1.1 Experiment 4

Experiment 4 (EXP4) "Decision making for motion planning when faced with roadworks, unmarked lanes and narrow roads with assistance from perception self-assessment" is an end-to-end experiment starting with the precise vehicle localization, by defining a semantic representation of the environment, and the motion prediction of dynamic objects in the scene. Particularly in the context of roadworks, unmarked lanes and narrow roads, the ego-vehicle performs a self-assessment by deciding whether to trust its perception system. Using a demo vehicle, the trajectory and motion planning of the ego-vehicle will be defined in real time. Further, the ego-vehicle's behavioural decision making will be tested by using long/short-term MOT/MOP in a cascaded integration approach in which the prediction estimation (other vehicles, VRUs, etc.) feeds the ego-vehicle's behavioural decision-making and vice-versa. Finally, the vehicle's control algorithms will be enhanced with a fail operational mode to track cases of positioning failure (T4.3). The different modules of this experiment will be tested in demo vehicles and in a simulator.

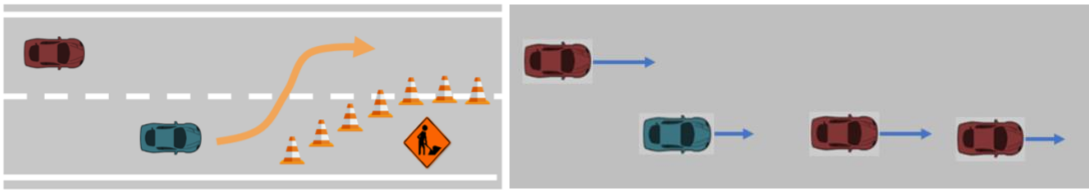


Figure 12a: EXP4 Graphical Representation (Ego-vehicle in green)

4.4.1.2 Experiment 5

Experiment 5 (EXP5) "Decision making for motion planning when entering a jammed highway" is similar to EXP4 with two main differences; the first is that there is no self-assessment of the ego-vehicle, and the second difference is that the motion planning involves path and speed planning as well as control tasks related to highway entrance.

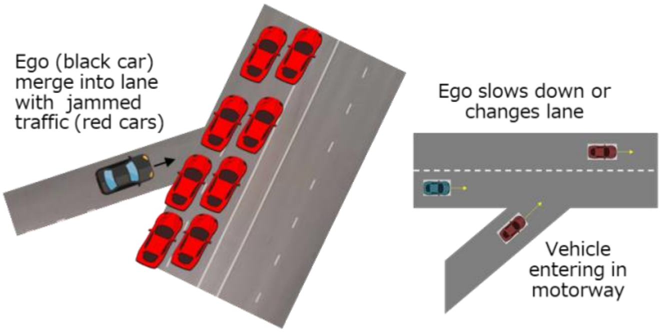


Figure 12b: EXP5 Graphical Representation (Ego-vehicle in black/green)

HITACHI is the leader of these Experiments and **CRF**, **TECN** and **WMG** are partners.

4.4.2 High-level Architecture and Interfaces

Figure 13 depicts the high-level architecture of EXPs 4&5. It is a direct subset of the Master Architecture (Figure 2). Description of the modules are as described in Section 0.

4.4.1 Detailed Architecture and Interfaces

For readability, the detailed architecture has been split into "Perception" and "Decision/Motion Planning and Control/Actuation".

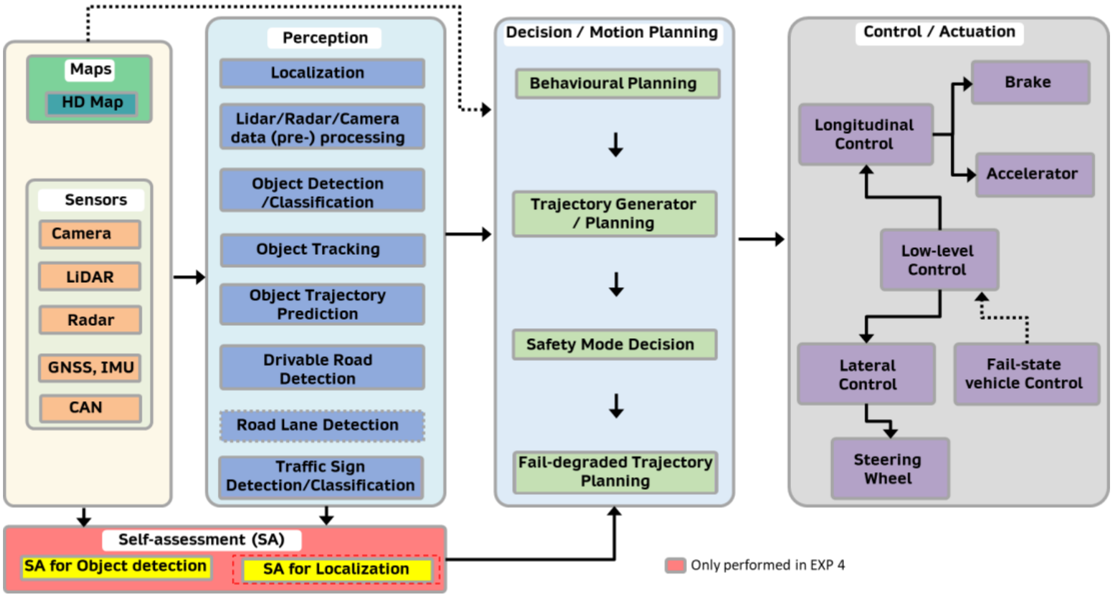


Figure 13: EXPs 4&5 high-level Full Stack Architecture and Interfaces

Perception

The detailed Perception architecture and interfaces is shown in Figure 14a. Main functions are described below:

- **Sensor Calibration**: The sensor calibration module will do an offline calibration from raw sensor data for cameras and the extrinsic calibration between different cameras, between different LiDARs and between camera and LiDAR. This module provides the calibration matrix for detection and classification tasks.
    - Intrinsic calibration: A calibration process to estimate the internal parameters of the sensor (camera) such as the focal length and lens distortion.
    - Extrinsic calibration: A calibration process to estimate the extrinsic parameters that contains the rotation matrix and translation vector between two coordinate systems.
- **Camera based 2D Object Detection**: The camera based 2D object detection module detects and classifies 2D objects from raw camera images. This module involves detecting (and classifying) **road lanes** and **traffic signs**.
    - YOLO[3]: stands for "You Only Look Once" is a real-time detection algorithm to detects and recognizes various objects in an image.

[3] https://docs.ultralytics.com/

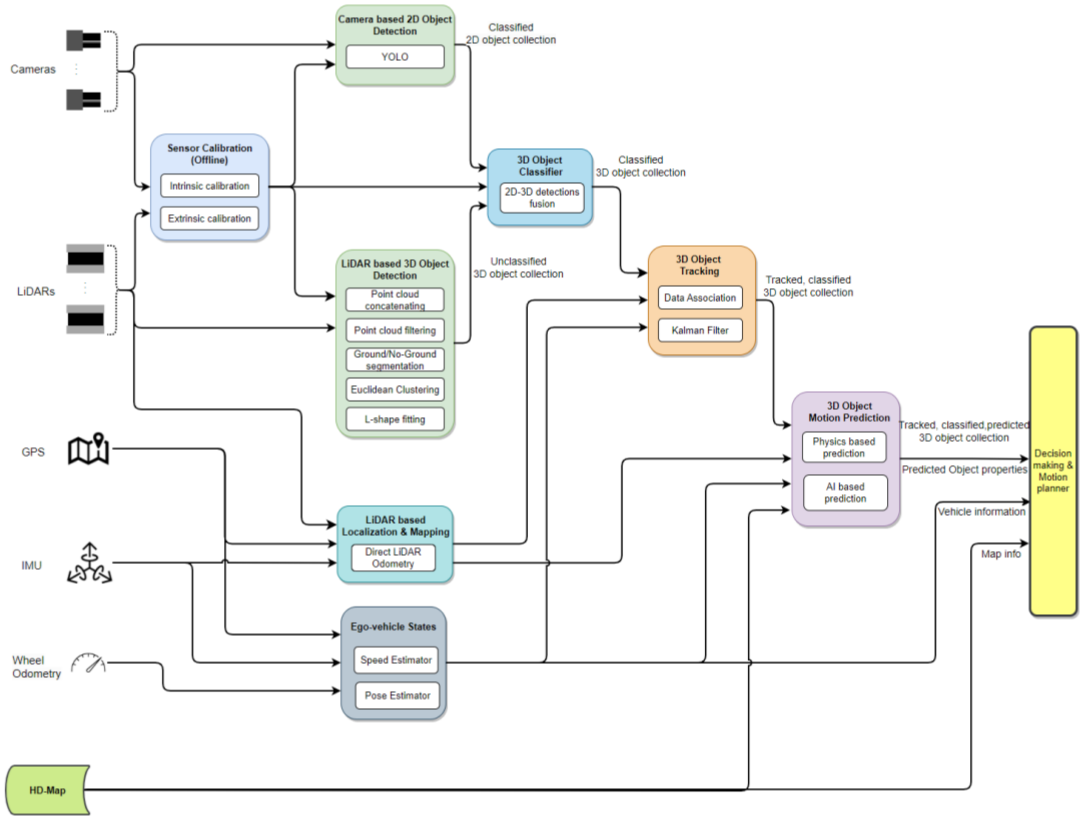


Figure 14a: EXPs 4&5 Detailed Full Stack Architecture and Interfaces (Perception)

- **LiDAR-based 3D Object Detection**: The LiDAR-based 3D Object Detection module clusters and detects 3D objects from concatenated and calibrated point cloud data from all LiDARs. The 3D object detection output is not classified at this stage.
    - Point cloud concatenating: This module concatenates and aligns in time and space the point cloud data from different LiDAR sensors.
    - Point cloud filtering: This module filters points that is out of interest or down sampling points.
    - Ground/No-Ground segmentation: This is component to segment the points correspond to ground and no-ground.
    - Euclidean Clustering: This module clusters or groups the points based on Euclidean distance.
    - L-shape fitting: The module fits or computes a bounding box which best fits an L-shaped cluster points.

- **3D Object Classifier**: The 3D object classifier module fuses the 2D detection output from camera-based algorithm and the 3D detection output from LiDAR-based algorithm to obtain the class of the 3D object. The extrinsic calibration provides the transformation matrix to fuse 3D and 2D object.
    - 2D-3D detections fusion: This module fuses 2D and 3D detection to classify the 3D detection.
- **3D Object Tracking**: The 3D object tracking module tracks over time the detected and classified 3D objects and outputs the dynamics (velocity and acceleration) of the objects. The core of the module are the Kalman filter and the data association to filter and associate objects between frames.
    - Data association: This module focuses on associating objects between frames.
    - Kalman filter: This is a Bayesian filtering algorithm for estimating state parameters in the presence of uncertainty (Gaussian noise).
- **3D Object Motion Prediction**: The 3D object motion prediction module predicts the future movement of the tracked objects based on their dynamical information and semantic information of the surrounding environment.
    - Physics based prediction: This is a motion prediction algorithm based on physical model.
    - Artificial Intelligence (AI)-based prediction: This is a motion prediction algorithm based on a machine learning model.
- **LiDAR based Localization & Mapping**: The LiDAR based localization and mapping module provides simultaneously the localization and mapping using LiDAR point cloud data.
    - Direct Lidar Odometry (DLO): This is an algorithm that simultaneously performs localization and mapping using the LiDAR point cloud.
- **Ego-vehicle States**: The ego-vehicle states modules provide all information related to ego-vehicle such as speed, pose, etc.
    - Speed estimator: This submodule estimates the speed of the ego-vehicle.
    - Pose estimator: This submodule estimates the pose of the ego-vehicle.

Perception

The sub-modules in this block have been described in Section 4.6.3, please refer to Section 4.6.3 for details.

Decision/Motion Planning

The detailed Decision/Motion Planning architecture and interfaces are shown in Figure 14b. The main functions are described below:

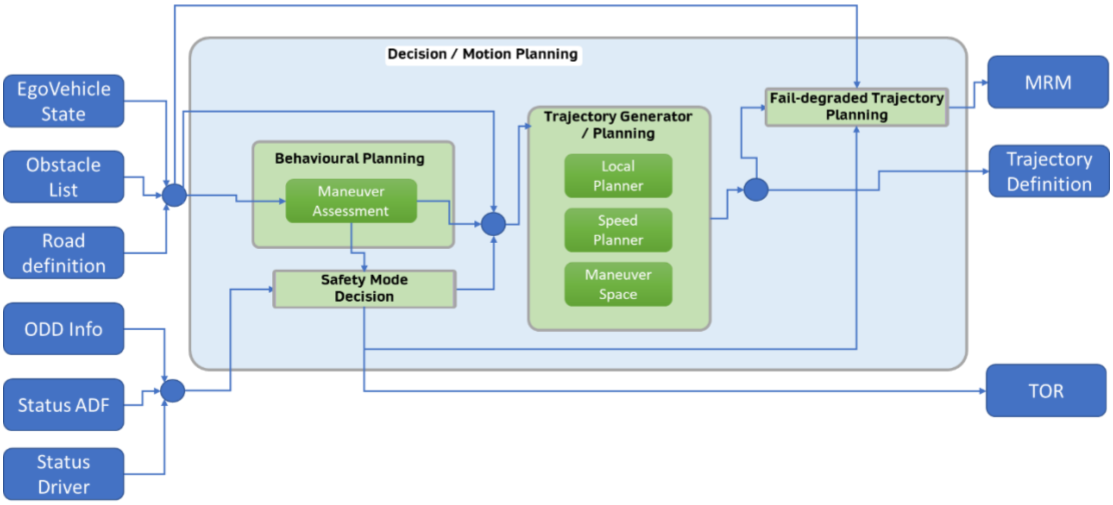


Figure 14b: EXPs 4&5 Detailed Full Stack Architecture and Interfaces (Decision/Motion Planning).

- **Behavioural Planning**: This software module is a part of the Decision/Motion planning, which will take care of high-level decision making in the form of a maneuver selection considering disturbances and unexpected behaviors from other agents. This software module can also assess the vehicle and driver capabilities in order to provide a transition control mechanism for each specific scenario.
  - Inputs:
    - Information from ego-vehicle (e.g., position, speed, acceleration, orientation, etc.).
    - Estimated state of Obstacles (e.g., position, speed, acceleration, orientation, type, id-tracking, prediction for other road participants, etc.).
    - Information on road layout (e.g., lane configuration, drivable space, rules, horizontal/vertical traffic signs, connectivity map).

        - Status of driver (e.g., level of distraction, impaired / ready, and so on).
        - Information of ODD (e.g., range within still valid).
        - Status of Automated Driving Function (e.g., level of confidence from perception platform, possible fail of sensors, etc.).
    - Outputs:
        - Manoeuvre selection/primitive for trajectory.
        - Decision for control transfer.

- **Trajectory Generator/Planning**: This software module is responsible for generating a feasible and safe trajectory in real time, taking as input a high-level decision or maneuver, and giving as output a series of waypoints for the low-level control of the vehicle. It will be subdivided in two parts; a **local planner**, which will define the path based on the maneuver and map definition, a **speed planner** which will define speed profiles to accomplish the maneuvers in a safe and comfortable manner.
    - Inputs:
        - Information from ego-vehicle (e.g., position, speed, acceleration, orientation, etc.).
        - Information from Obstacles (e.g., position, speed, acceleration, orientation, type, id-tracking, etc.).
        - Information on road layout (e.g., lane configuration, drivable space, rules, horizontal/vertical traffic signs, connectivity map).
        - Information from Primitive trajectory or manoeuvre selection from Behavioural planner.
    - Outputs: Trajectory definition (speed reference and waypoints)

- **Safety Mode Decision**: This module can imply the decision about "who is in charge of what", taking into account:
        - The current ODD limits of the Automated Driving Function (including the scenario complexity).

    - The status of the driver (e.g., if s/he can respond to a give request).
    - The status of the system (e.g., level of confidence from the perception platform).
  - Inputs:
    - Information of ODD (e.g., range within still valid).
    - Status of Automated Driving Function (e.g., level of confidence from perception platform, possible fail of sensors, etc.).
    - Status of driver (e.g., level of distraction, impaired / ready, and so on).
  - Outputs:
    - Best action to be performed
    - Take Over Request issued to driver
    - Agent performing the action

Different methods can be used, such as Fuzzy Inference System, or Model Predictive Control (MPC) (tuned by a more sophisticated State-of-the-Art (SoA) algorithm, such as Reinforcement Learning).

- **Fail-degraded Trajectory Planning**: This module provides the Minimum Risk Manoeuvre (MRM), if necessary, which is strictly related to the demo-vehicle and must be attainable by that vehicle.
  - Inputs:
    - Information from Safety-mode Decision (what to do, from whom, when).
    - Information on road ahead (e.g., exit, end of motorway, etc.).
    - Information on road layout (e.g., presence of emergency lane on the right).
    - Information on obstacles (e.g., stationary / slower object(s) in front of the ego-vehicle, if the emergency lane is free, and so on).

- Outputs: Minimum Risk Manoeuvre to be executed.

The minimum risk maneuver can be an emergency stop in the driving lane or a safe lane change in the emergency lane (if any).

Control/Actuation

The goal of this module is to provide the control inputs to the actuators, to move the prototype vehicle.

The responsibility of this module is of the single demo-owner, due to its specificity and relation with the safety aspects.

Generally, in a "normal situation", the outputs of the "Trajectory Generator / Planning" block are provided to the "Low-level Control" block.

In "abnormal situation" (namely, low confidential level from perception platform, fault of sensors, etc.), the outputs of the "**Fail-degraded Trajectory Planning**" block goes to the "**Fail-state vehicle Control**", which is responsible to actuate the minimum risk manoeuvre.

In both cases, the "**Low-level Control**" block is responsible to provide the inputs both for the longitudinal and lateral control, driving the respective actuators:

- accelerator and braking pedals for the longitudinal control
- the steering wheel for the lateral control.

The exact and specific signals to/from each module and related block will be defined later on in the project.

The above control description is tailored to the CRF demo vehicle for EXPs 4&5. Another variant of the above Control/Actuation description with another demo vehicle (Renault Twizy) of Tecnalia, a partner of EXPs 4&5 is as follows:

- The platform has been fitted with a PLC and two servo motors.
    - The PLC handles the signals from the vehicle intelligence to the actuators via CAN bus.
    - A servo motor has been fitted to actuate over the brakes.
    - A servo motor has been fitted to actuate over the steering wheel.
- An industrial computer which oversees decision making and control sends signals to the PLC via CAN bus.

The input to the actuators is managed via a "low-level control" block, which in turn depends on the decision module, and will alter its output upon an evaluation of the risk.

4.5 Experiment 6

4.5.1 Description

Experiment 6 (EXP6) "Small object detection at a far range in adverse weather conditions" will demonstrate a situation/scenario where the ego-vehicle approaches a static object (e.g., debris) present on the ego-lane at a high speed. The visibility is deteriorated due to heavy rain/fog/snow during night/day. The vehicle should perform a full stop to avoid a collision or drive over the object safely.
APTIV is the leader and sole participant in this experiment.

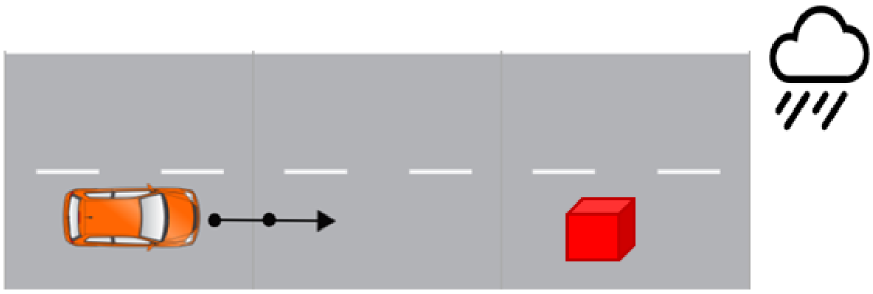


Figure 15: EXP6 Graphical Representation

4.5.2 High-level Architecture and Interfaces

Figure 16 depicts the high-level architecture of EXP6. It is a direct subset of the Master Architecture (Figure 2). Description of the modules are as described in Section 0.

4.5.1 Detailed architecture and Interfaces

Figure 17 shows the detailed architecture and inter-relation between modules for the realization of EXP6 with a focus on Perception. The main items in the architecture are explained as follows:

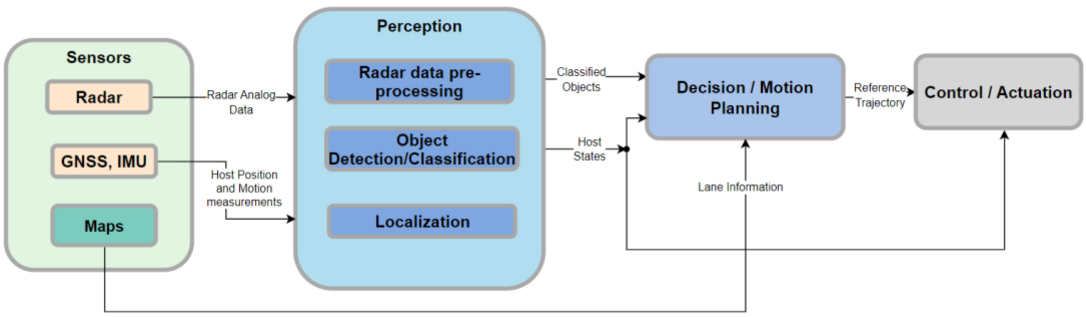


Figure 16: EXP6 high-level Full Stack Architecture and Interfaces

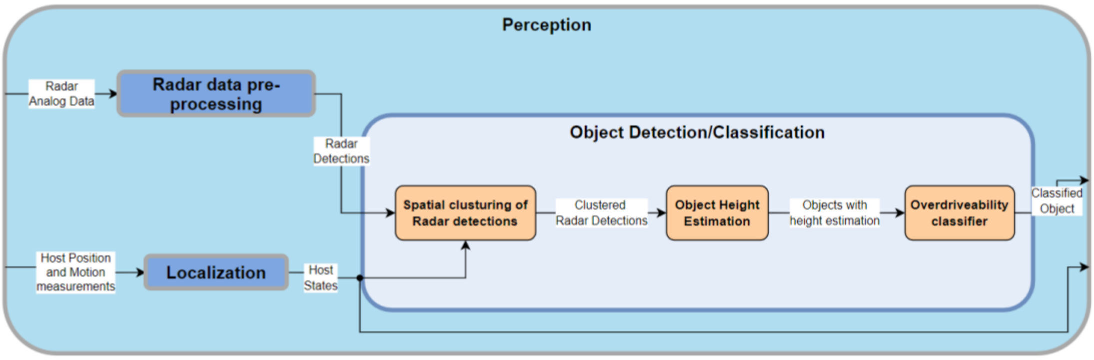


Figure 17: EXP6 Detailed Full Stack Architecture and Interfaces

Sensors: EXP6 will primarily rely on Radar, GNSS/IMU, and Maps. The Radars output analog data based on reflections from objects in the field of view and the GNSS system outputs the host vehicle's position and motion measurements.

Perception: The perception module in EXP6 detects and classifies objects in the driving lane and classifies them as over-drivable or non-over-drivable. It uses three sub-blocks, namely, Radar data pre-processing, Localization, and Object detection/Classification.

- **Radar data pre-processing**: This module receives the analog data from the radars and processes them to output Radar detections – which are discrete points in space with information about the range, range rate, and azimuth angle of the detection point.
- **Localization**: This module receives the Host position and motion measurements from the GNSS system and uses internal models and filters to output a more accurate Host state estimate.
- **Object Detection/Classification**: This module uses Radar detections and Host state estimates to detect objects in the field of view of the sensors and to classify them as over-drivable or non-over-drivable. This is done in three steps:
  - Spatial clustering of Radar detections: The radar detection points are first clustered based on a region of interest. This helps in filtering out irrelevant detections for the context of the driving scenario.
  - Object Height Estimation: The clustered radar detections are used to estimate the height of each cluster using machine learning methods.

- Over-drivability classifier: Based on the estimated heights of the detection clusters, a decision is made on whether the object associated with the cluster is over-drivable or not.

- **Decision/Motion Planning:** (Not developed in EXP6 but presented here for completeness) The Decision/Motion Planning module receives the classified objects and Host state estimates from the Perception system and the lane information from maps to determine the desired maneuver. In the context of EXP6, this can either be an adaptation of vehicle speed to drive over an object or to come to a complete stop. Based on this maneuver a reference trajectory is sent to the control/actuation module.

- **Control/Actuation:** (Not developed in EXP6 but presented here for completeness) The Control/Actuation module receives the reference trajectory from the Decision/Motion Planning and is responsible for the lateral and longitudinal motion control of the vehicle.

4.6 Experiment 7

4.6.1 Description

Experiment 7 (EXP7) “Localization/perception self-assessment for advanced ACC and other vehicles’ behavior prediction under adverse weather or adverse road conditions” focuses on the development of a self-assessment mechanism for perception under adverse weather conditions in urban or motorway driving as well as the development of an integrity monitoring mechanism for perception-based localization. For urban driving, the automated driving function can be activated and an integrity monitoring mechanism for GNSS-based localization shall be implemented/activated alongside too. Those mechanisms should reliably indicate the point in time when the localization of the ego-vehicle and/or its perception must not be trusted. In that case, a handover request to the human driver will be activated. Another objective (not related to the self-assessment objective) is to study the effects of adverse weather conditions on a perception module performing other vehicles’ behavior prediction.

Figure 18: EXP7 Graphical Representation (Ego-vehicle in black)

4.6.2 High-level Architecture and Interfaces

Figure 19 depicts the high-level architecture of EXP7. It is a direct subset of the Master Architecture (Figure 2). The description of the modules is presented in Section 0.

4.6.3 Detailed Architecture and Interfaces

Figure 20 shows the detailed architecture and the inter-relation between various modules for the realization of EXP7 with a focus on (i) an introspection mechanism for perception and (ii) an integrity monitoring mechanism for GNSS-based localization and perception-based localization to the leading vehicle for the purpose of ACC. The main items in the architecture are explained as follows:

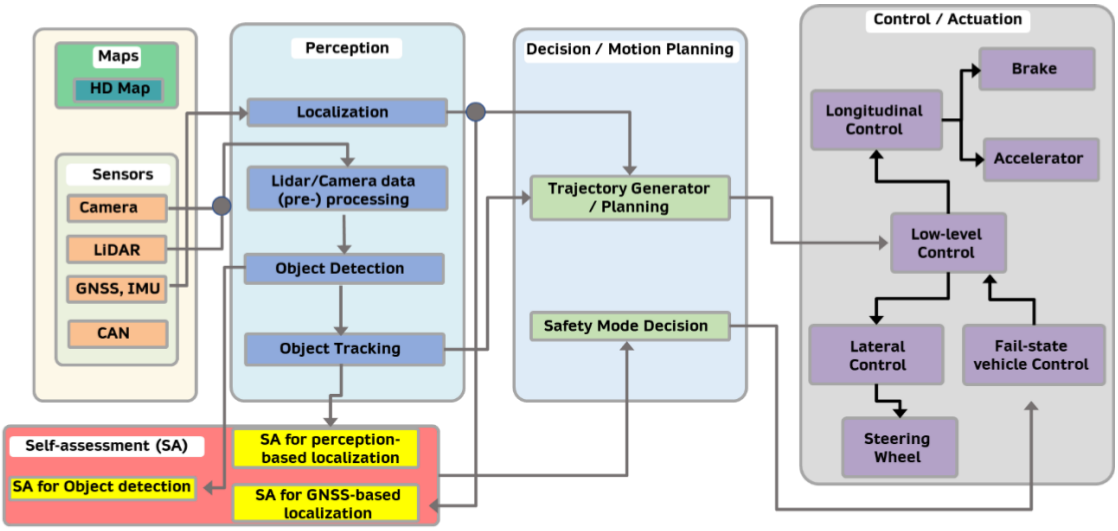


Figure 19: EXP7 high-level Full Stack Architecture and Interfaces

- **Localization**: This module involves the positioning of the ego vehicle using GNSS.
    - For urban driving, the latitude and longitude coordinates are fed to the integrity monitoring mechanism **SA for GNSS-based localization** that uses Receiver Autonomous Integrity Monitoring (RAIM).
- **Lidar/Camera pre-processing**: This module involves using either the LiDAR point cloud or cameras to detect static and dynamic objects around the ego-vehicle.
- **Object detection**: This includes the classification and localization of the surrounding static and dynamic objects using Yolo (for camera-based 2d object detection) or Clustering/CenterPoint (for lidar-based 3d object detection).

- **SA for object detection**: In this module, raw activation patterns (last layer of the backbone CNN indicated by the dashed line) are pre-processed and fed to the SA mechanism for introspecting (e.g., RESNET for features extraction and a fully-connected neural network for classification) perception errors.
- **Object tracking:** This module includes tracking of the detected objects using autoware-ai IMM-UKF-PDA track.
- **SA for perception-based localization:** This involves identification of the leading vehicle, estimating its distance, and monitoring its integrity
- **Trajectory generation:** This module includes a pure pursuit controller to follow pre-defined waypoints and a bespoke ACC (car following model) in case a lead vehicle is detected.
- **Safety mode decision:** This module assesses whether the perception quality and/or the localization integrity (GNSS-based or perception-based) is poor and triggers a takeover request to the human driver if it returns positive.

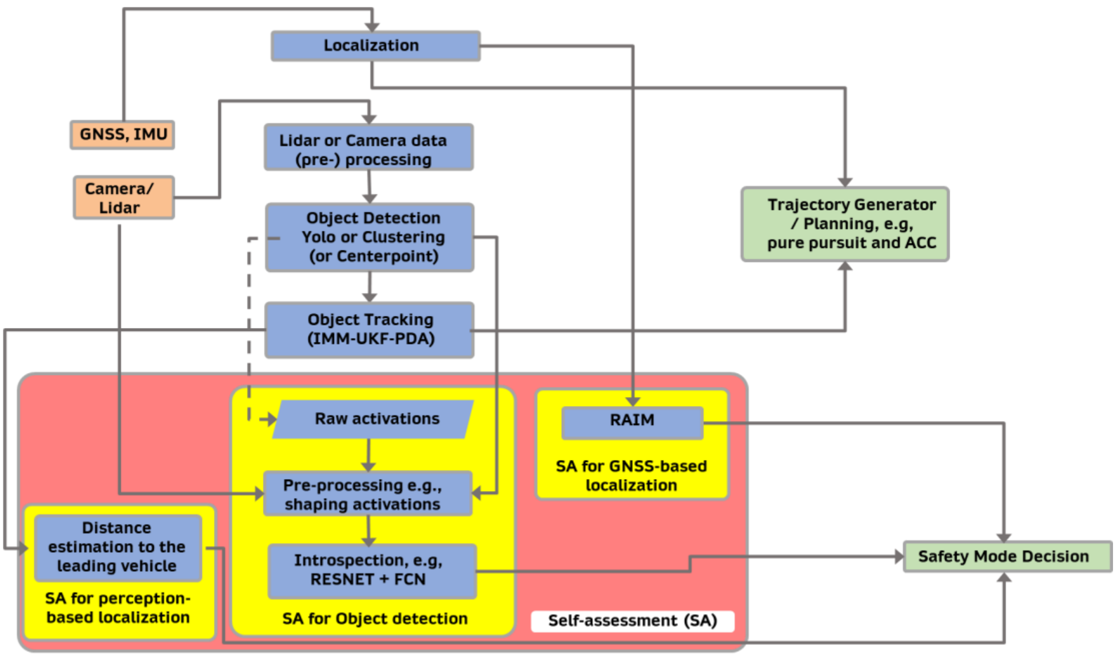


Figure 20: EXP7 Detailed Full Stack Architecture and Interfaces

4.7 Experiment 8

4.7.1 Description

Experiment 8 (EXP8) "Driving minor road under adverse weather conditions including perception self-assessment" is similar to EXP7 but in slow-speed roads instead of motorways (Figure 21). Driving on minor roads (<40km/h) under adverse weather conditions. ADS adapts to weather conditions. (Simulation or controlled conditions can be used to produce the conditions)

The low atmospheric visibility in adverse weather conditions like fog, snow and rain reduces the maximum viewing distance of LIDAR sensors. This in turn decreases the object detection and localization performance and can cause safety hazards.

Weather conditions have effect to sensing and therefore to the perception and localization of automated driving system. This EXP provides possibility to evaluate the on-board visibility-based localization performance estimate. Safe vehicle control is necessary in case of worsening of weather conditions and fail-safe behavior in case of exiting the ODD completely due to extreme weather.

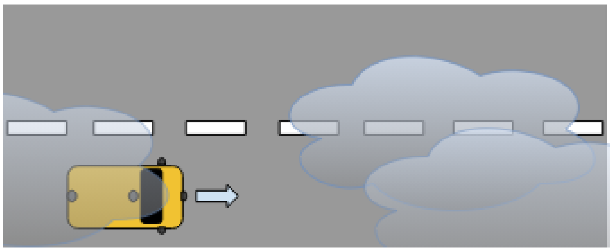


Figure 21: EXP8 Graphical Representation

4.7.2 High-level Architecture and Interfaces

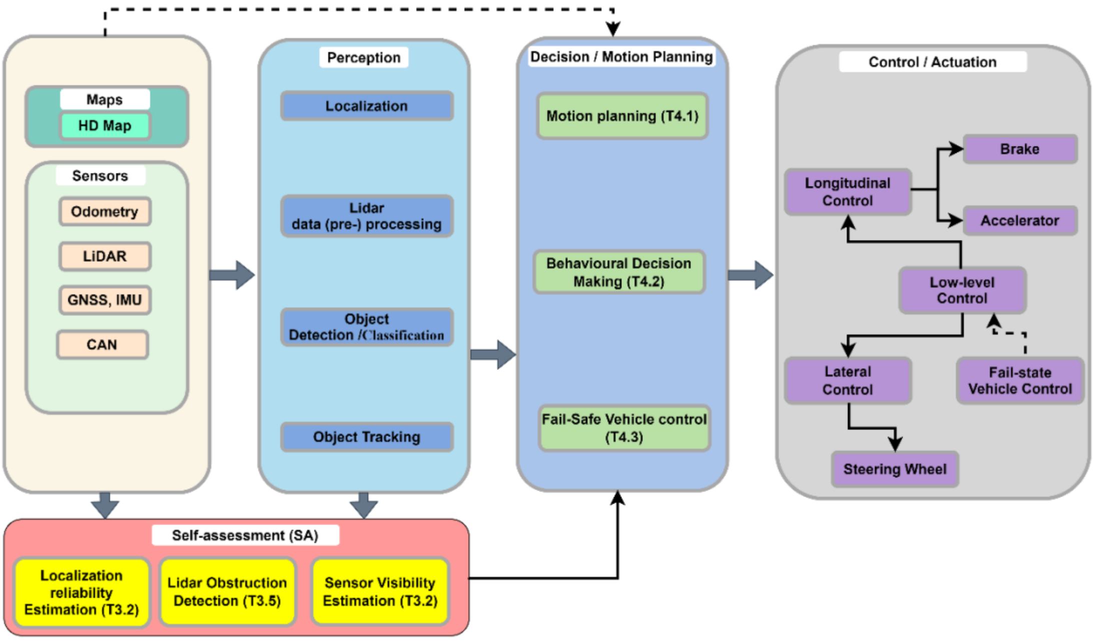


Figure 22: EXP8 high-level Full Stack Architecture and Interfaces

Figure 22 depicts the high-level architecture of EXP8. It's a direct subset of the Master architecture (Figure 2). Description of the modules are described in Section 0.

4.7.3 Detailed Architecture and Interfaces

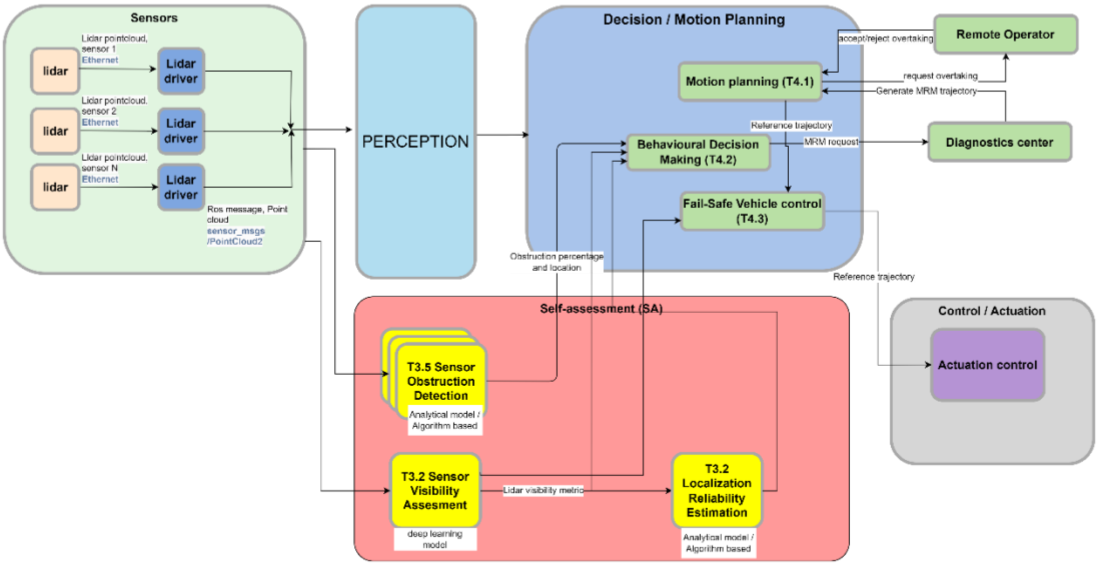


Figure 23: EXP8 Detailed Full Stack Architecture and Interfaces

- **Sensors:** EXP8 will primarily rely on LIDAR, GNSS/IMU and Maps.
- **Perception:** Perception module provides its data to all the submodules of the Decision/Motion Planning block. Motion planning, Behavioural Decision-making and Fail-safe vehicle control all receive this data.
- **Motion Planning:** Motion planning module generates trajectories from the perception data it receives, but it also handles asking for permission for overtaking from the remote operator, and then generating the overtaking trajectory. It can also receive commands from the diagnostics center to create a MRM trajectory if triggered by Behavioural Decision making.

Interface descriptions:

- Input:
  - Raw lidar point cloud data
  - Overtaking confirmation
  - MRM trajectory request
- Output:
  - Reference trajectory

- Overtaking request

- **Behavioral Decision Making**: This module tracks the state of the vehicle and its sensors from the self-assessment modules submodules Sensor obstruction detection and localization Reliability estimation. Using this data the module decides if the vehicle is operating inside the ODD. if the vehicle is operating outside the ODD it sends a message to the diagnostics center to generate an MRM trajectory, which then commands motion planning to create an MRM trajectory for actuation control.

Interface descriptions:

- Input:
  - Obstruction amount (in percentage) and location (azimuth angle),
  - 2D vector representing estimated visibility along azimuth and associated uncertainty
  - Scalar value for expected localization error
- Output: MRM request

- **Fail-Safe Vehicle control** This module provides the vehicle with fail-safe capabilities where the module independently tracks the validity of data received from the sensors and can then adjust the reference trajectory for a lower speed based on the actual ODD situation outside.

Interface descriptions:

- Input:
  - Current visibility estimate and associated confidence value
  - Reference trajectory
- Output: Speed adjusted reference trajectory

- **Sensor Obstruction Detection** This module assesses the current state (field of view) of the lidar sensor. In essence, it checks the field of view for any blockages due to dirt, frost or snow. Accordingly, it sends the obstruction amount (in percentage) and location.

Interface descriptions:

- Input: Raw lidar point cloud data
- Output: Obstruction amount (in percentage) and location (azimuth angle)

- **Sensor Visibility Assessment** This module assesses the current state of visibility in close proximity. It essentially provides an estimate for atmospheric visibility around the vehicle and also computes the associated confidence values.

Interface descriptions:

- Input: Raw lidar point cloud data
- Output: 2D vector representing estimated visibility along azimuth and associated uncertainty

- **Localization Reliability Estimation** This module estimates the state of localization performance for current weather condition. It essentially obtains the expected localization error value by checking the given visibility value in the lookup table.

Interface descriptions:

- Input: 2D vector representing estimated visibility along azimuth and associated uncertainty
- Output: Scalar value for expected localization error

- **Remote Operator** Remote operator approves or denies overtaking requests from the vehicle, making the decision for those.

Interface descriptions:

- Input: Overtaking request
- Output: Overtaking confirmation

- **Diagnostics Center** This module triggers MRM, when requested by the behavioural decision making module. It accordingly request the motion planning module to generate the MRM trajectory.

Interface descriptions:

- Input: MRM request
- Output: MRM trajectory request

5. Conclusion & Future Work

In this document, the full stack architecture and interfaces of the EVENTS project are defined. The architecture includes both a high level (master) architecture of the project as well as the individual architectures of the 8 experiments, which are briefly described in Chapter 2 and in detail in Deliverable D2.1 “User and System Requirements for selected Use-cases” [1].

This document will guide all the technical development and integration for the rest of the project, serving in particular as input of

- Task T2.4 and subsequently of Deliverable D2.3 “Vehicle System Hazard Analysis & Risk Assessment”,
- Work Packages 3 (Perception and self-assessment) & 4 (On-board decision making for fail-safe vehicle motion), i.e., during the development phase, in which the architecture will also be fine-tuned, and
- Work Package 5 (System integration and safety compliance).

It should be noted that the software architecture will be defined at a later stage during the development in WP4 & WP5.

References

[1] EVENTS Deliverable D2.1: User and system requirements for selected use cases (2023)