D7.2

System prototype demonstration

Project Acronym	TransAID
Project Title	Transition Areas for Infrastructure-Assisted Driving
Project Number	Horizon 2020 ART-05-2016 – GA Nr 723390
Work Package	WP7 Real-World Prototype
Lead Beneficiary	German Aerospace Center (DLR)
Editor / Main Author	Julian Schindler DLR
Reviewer	Robbin Blokpoel DYN
Dissemination Level	PU
Contractual Delivery Date	30/06/2019 (M22)
Actual Delivery Date	31/08/2019
Version	v1.0

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 723390.

Document revision history

Version	Date	Comments
v0.1	2019-07-03	Initial draft version
v0.2	2019-08-07	Added vehicle and road side architecture
v0.3	2019-08-14	Added first iteration scenarios and feasibility structure
v0.4	2019-08-26	Included several components' content, sent for review
v0.5	2019-08-29	Included several reviewer comments and updated several images.
v0.6	2019-08-30	Second review performed.
v1.0	2019-08-31	Final editing and submission.

Editor / Main author

Julian Schindler (DLR)

List of contributors

Robert Markowski (DLR), Daniel Heß (DLR), Daniel Wesemeyer (DLR), Clarissa Böker (DLR), Baldomero Coll Perales (UMH), Miguel Sepulcre (UMH), Javier Gozalvez (UMH), Michele Rondinone (HYU), Dominik Matheis (HYU)

List of reviewers

Robbin Blokpoel (DYN)

Dissemination level:

■ PU : Public

☐ RE : Restricted to a group specified by the consortium (including the Commission Services)

☐ CO : Confidential, only for members of the consortium (including the Commission Services)

Table of contents

Executive Summary

This deliverable is a direct successor of Deliverable 7.1 [1], which has introduced all vehicles, test tracks and used hardware, and also proposed system architectures of the different used components. D7.1 has also introduced several system requirements for each component and for each scenario, which has to be implemented.

D7.2 now shows the results of the first project integration phase. The system implementation is described for the different components of the infrastructure part as well as for the vehicle part. It is shown how both parts communicate in the real world during the first project iteration by presenting the used ASN.1 message definitions (in the Annex) and details about the communication software.

Furthermore, a feasibility assessment has been performed by the project partner HMETC. For this, each scenario has been divided into test cases, which have been implemented in the real world prototypes, and demonstrated on a test track in northern Germany.

Each test case is linked to related requirements set up in D7.1. During the feasibility assessment, the compliance with all requirements has been checked. In addition, the overall "look and feel" of the prototype and the performance in each test case has been rated and described.

In summary, most of the requirements were met. Nevertheless, some deviations have been found. Most of those deviations will be fixed during the second iteration of the project, but there were also some minor points, which need to be reformulated during the second iteration. These points include some identified weaknesses and some needed re-interpretations of existing fields in the used messages.

Altogether, it could be shown that the TransAID ideas can be put into real-world to help future automated vehicles to better cope with possible threats and to gain higher performance on the road.

1. Introduction

1.1 About TransAID

As the introduction of automated vehicles (AV) becomes feasible, even in urban areas, it will be necessary to investigate their impacts on traffic safety and efficiency. This is particularly true during the early stages of market introduction, when automated vehicles of different SAE levels, connected vehicles (able to communicate via V2X) and conventional vehicles will share the same roads with varying penetration rates.

There will be areas and situations on the roads where high automation can be granted, and others where it is not allowed or not possible due to missing sensor inputs, high complexity situations, etc. At these areas, many automated vehicles will change their level of automation. We refer to these areas as “Transition Areas”.

TransAID develops and demonstrates traffic management procedures and protocols to enable smooth coexistence of automated, connected, and conventional vehicles, especially at Transition Areas. A hierarchical approach is followed where control actions are implemented at different layers including centralised traffic management, infrastructure, and vehicles.

First, simulations are performed to examine efficient infrastructure-assisted management solutions to control connected, automated, and conventional vehicles at Transition Areas, taking into account traffic safety and efficiency metrics. Then, communication protocols for the cooperation between connected/automated vehicles and the road infrastructure are developed. Measures to detect and inform conventional vehicles are also addressed. The most promising solutions are then implemented as real world prototypes and demonstrated at a test track and during the second iteration possibly on public roads. Finally, guidelines for advanced infrastructure-assisted driving are formulated. These guidelines also include a roadmap defining activities and needed upgrades of road infrastructure in the upcoming 15+ years in order to guarantee a smooth coexistence of conventional, connected, and automated vehicles.

1.1.1 Iterative project approach

The infrastructure-assisted management solutions are developed and tested in two iterations, each taking half of the project total duration. During the first iteration, the focus is on studying aspects of transition of control (ToC) and transition areas (TAs) through basic scenarios. This implies that realistic models for automated driving (AD) and ToC need to be developed and/or adopted. Using the basic scenarios, it is possible to run many simulations and focus in detail on the relatively new aspects of ToC, Transition Areas (TAs) and measures mitigating negative effects of TAs. The goal of the first iteration is to gain experience in modelling, simulation and real world implementation with all aspects relevant to TAs and the mitigating measures.

During the second iteration, that experience is used to improve/extend the measures while at the same time increasing the complexity of the scenarios and/or selecting different (more complex) scenarios. Another possibility under consideration is the combination of multiple basic scenarios into one new more complex use case.

1.2 Purpose of this document

As a successor document of D7.1 [1], this deliverable is describing all implementation actions for the real-world prototype, which have been taking place during the first project iteration of TransAID. In addition to this, a feasibility assessment of the developed prototype has been done by the project partner HMETC. Therefore, each TransAID service and related scenario (see D2.2 [2]) has been transferred into test cases. The requirements for the different scenarios, which have already been described in D7.1, are now related to the test cases and the compliance is discussed.

Besides describing the procedures, a goal of this deliverable is to investigate which parts of the message definition (see D5.1 [3] and D5.2 [4]) and of the TransAID traffic management measures (see D4.2 [5]) need to be adapted so that the system is not only performing in simulations (see D6.2 [6]) but also in the real world.

1.3 Structure of this document

This deliverable is first describing the prototype architectures for the used vehicles (section 2.1) and the used road side infrastructure components (section 2.2). Section 3 describes the performed feasibility assessment, including all test case descriptions and results per test case.

Besides the conclusion in section 4, this deliverable also contains the ASN.1 message definitions of the used messages during the real world assessment in the Annexes A and B.

1.4 Glossary

Abbreviation/Term	**Definition**
ACC	Adaptive Cruise Control
AD	Automated Driving
ADAS	Advanced Driver Assistance Systems
AV	Automated Vehicles (without cooperation abilities)
C-ITS	Cooperative Intelligent Transport Systems
C2C-CC	Car2Car Communication Consortium
CAM	Cooperative Awareness Message
CAV	Cooperative Automated Vehicle
CPM	Collective Perception Message
CV	Cooperative Vehicle
DENM	Decentralised Environmental Notification Message
DX.X	Deliverable X.X
ERTRAC	European Road Transport Research Advisory Council
HMI	Human Machine Interface
ITS	Intelligent Transport System
ITS-G5	Access technology to be used in frequency bands dedicated for European

	ITS
LOS	Level Of Service (from Highway Capacity Manual)
LV	Legacy Vehicle
MCM	Manoeuvre Coordination Message
MRM	Minimum Risk Manoeuvre
RSI	Road Side Infrastructure
RSU	Road Side Unit
SAE	Society of Automotive Engineers
SUMO	Simulation of Urban MObility
TA	Transition area
TM	Traffic Management
ToC	Transition of Control
TransAID	Transition Areas for Infrastructure-Assisted Driving
V2I	Vehicle-to-infrastructure (communication)
V2V	Vehicle-to-vehicle (communication)
V2X	Vehicle-to-anything (communication)
VMS	Variable Message Signs
WP	Work Package

2 Prototype architecture

In the following, the final prototype of the first project iteration is described. This section is based on section 4 of D7.1, and only adds more details to it.

2.1 Vehicles

During the tests performed a set of vehicles is used, including Cooperative Automated Vehicles (CAVs), Cooperative Vehicles without automation functionality (CVs) and legacy vehicles (LVs).

All CAVs and CVs are briefly described in the following.

2.1.1 CAVs

During the first iteration, two CAVs have been used, DLR's electric Volkswagen Golf "FASCarE" and DLR's hybrid Volkswagen Passat "ViewCar2". As both are from DLR, the internal setup is similar in both cars, with only minor differences in terms of used hardware revisions as the ViewCar2 is newer.

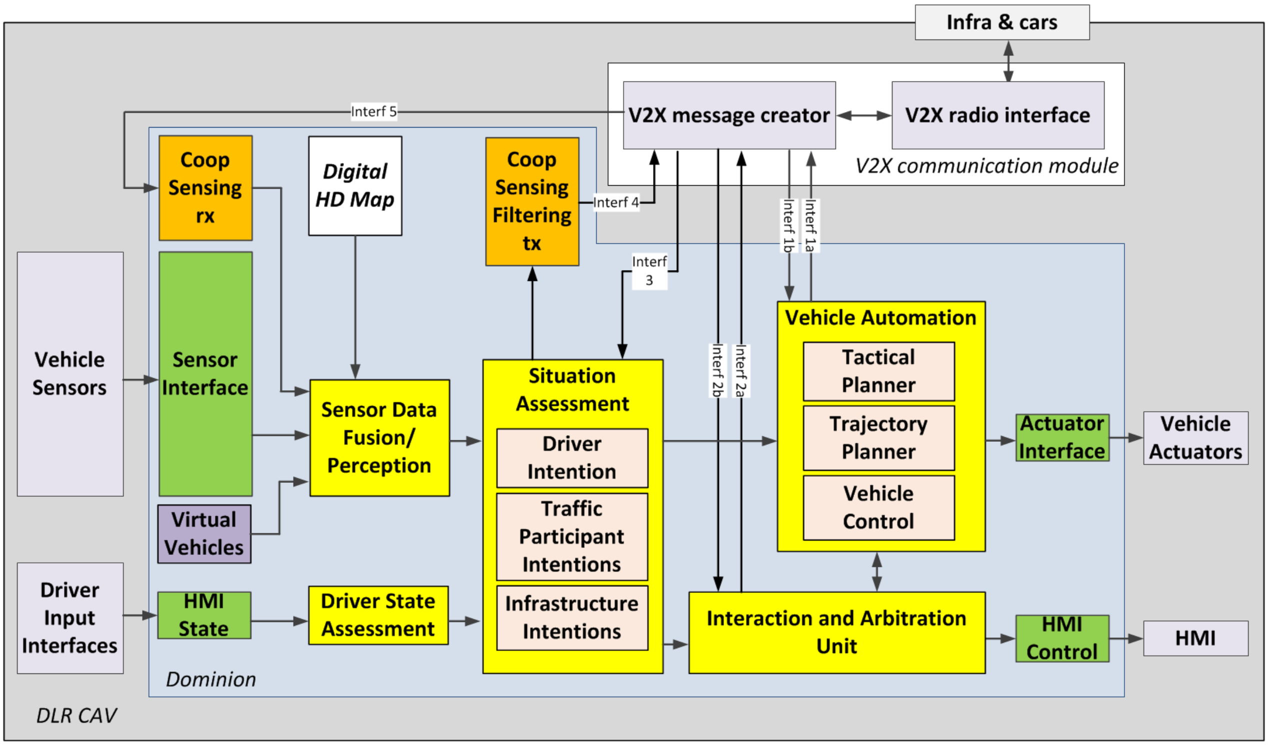


Figure 1: Initial CAV architecture

The CAVs basically follow the architecture shown in Figure 1 and described in D7.1. Only the component "Tactical Decision" has been renamed to "Tactical Planner", and "Trajectory Planning" has been renamed to "Trajectory Planner".

In the following, details about the sensors, sensor data fusion, vehicle automation and communication are given, which have been used during the first project iteration.

2.1.1.1 Sensors and Sensor Fusion

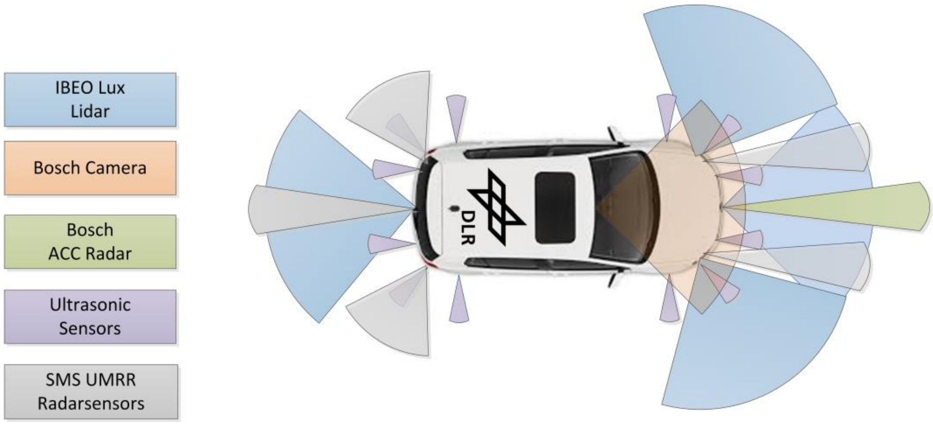


Figure 2: Sensor coverage of the FASCarE

As shown in Figure 2 in the example of the FASCarE, both of the research vehicles are equipped with multiple Ibeo laser scanners in the front and rear of each vehicle. The laser scanners are connected via Ethernet and integrated with the robot operating system (ROS). This is an open-source middleware framework comprising drivers for devices, message passing between processes implemented as nodes of a graph architecture, or implantations of frequently used functionalities (More information can be found on the ROS web page[1]). The objects determined from the laser scanner points are sent via a custom interface between the ROS framework to the Dominion framework [7] of the automation. Before the detected and tracked objects are passed on, they are also fused with the received CAMs. The current fusion iterates over the most resent CAM message, whenever new tracking results are received from the laser scanners, attempting to find the corresponding laser scanner object for each V2X object. If a correspondence can be established, the laser scanner object receives an additional tag and the V2X object is disregarded; otherwise the V2X object is appended to the list of detected objects. This way all unobserved V2X objects are added to the laser scanner tracking results. In the future a more elaborate fusion scheme based on covariance intersection will be implemented. This implementation will also include the fusion of received CPMs. This will be included in the revised version of this deliverable, to be submitted in M36.

2.1.1.2 Vehicle Automation

The planning and decision making modules for the TransAID CAVs have been implemented with the help of a vehicle automation library proposed in [8]. Accordingly, the CAV decision making is based on the four steps of environmental data aggregation, goal oriented data abstraction in so called views, manoeuvre planning and manoeuvre selection. Environmental data is received from the Sensor Data Fusion/Perception block in the form of the estimated ego state, static obstacles perceived by the vehicles laser scanners, traffic participant information consolidated from CAM, CPM and laser scanners, as well as road geometric and topological data from an HD digital map and a navigation component. The environmental data is abstracted in LaneFollowing-, LaneChange- and SafetyConstraintViews. The views allow formulating constraints for specific manoeuvre planning tasks. Each planned manoeuvre is rated by several different cost metrics. Decision making consists of selecting an appropriate, feasible and low-cost manoeuvre for execution by the Vehicle Control module. In addition to previous solutions, the project specific requirements relating to

[1] https://www.ros.org/

vehicle-driver, vehicle-infrastructure and vehicle-vehicle interactions are fulfilled by augmenting environmental data, manipulating constraint generation for the manoeuvre planners at the level of the goals of certain views and adjusting cost metrics.

As shown in Figure 1, the Tactical Planner component responsible for manoeuvre planning and plan selection receives input from the following components: Sensor Data Fusion/ Perception provides a list of static and dynamic objects and traffic participants. Map Provider sends geometric information about roads in the vicinity of the ego vehicle's current position to the Tactical Planner. This enables the Tactical Planner to maintain an up-to-date, local subset of the HD map. At the same time, the Map Provider serves the purpose of decoupling the Tactical Planning component from the source of the geometric road information: The pre-defined map can be replaced by sensor detections of lane border markings. The Navigation component sends lane-specific navigation information to the Tactical Planner. A cost-to-go is provided for every individual lane in order to evaluate the utility of lane changes. The communication module directly interacts with the Tactical Planner to support vehicle-to-vehicle manoeuvre coordination and to address lane- and speed-advice from infrastructure-to-vehicle communication directly on the impacted tactical level. The Tactical Planner generates and selects viable manoeuvres for execution and sends the according vehicle trajectories to the Vehicle Control component, which in turn choses control inputs (steering angle and acceleration) to minimize deviation from the trajectories.

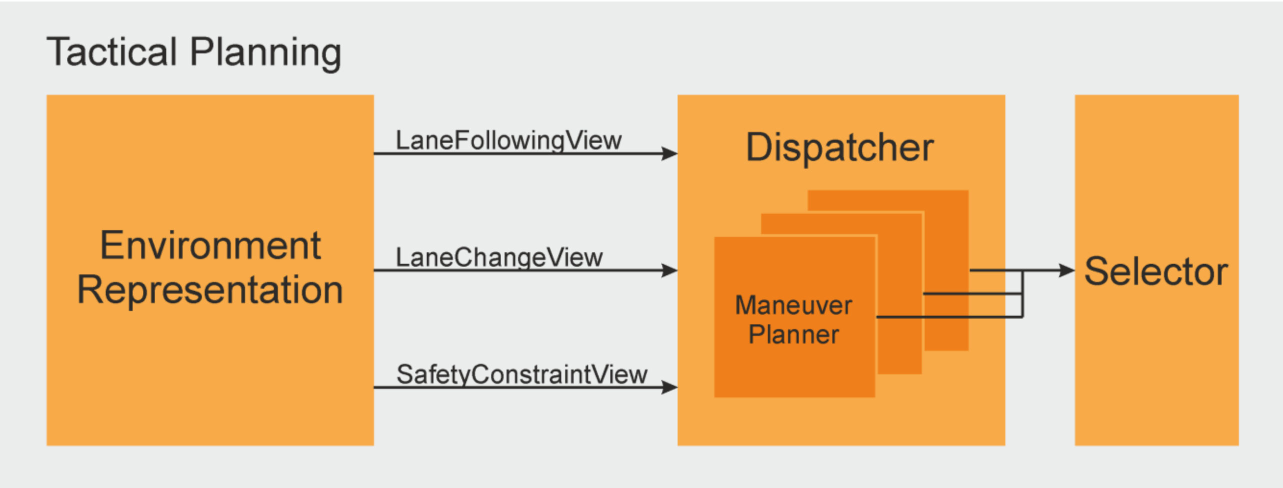


Figure 3: Tactical Planning with sub-components

The Tactical Planning component in turn consists of the following sub-components (fig. 7): Environment Representation aggregates data and generates LaneFollowing-, LaneChange- and SafetyContraintViews. Several instances of Manoeuvre Planner convert constraints specified by the views into concrete trajectories. The Dispatcher sub-component defines goals and convex constraint regions for Manoeuvre Planner instances and selects instances for plan computation. The selector component determines cost metric values and finally selects a manoeuvre for execution. The solution set of the domain is non-convex (for example distinct gaps in traffic, lane selection) and a cost function modelling the desirable behaviour can be non-linear and complicated. As a computationally efficient approximation of the globally optimal solution under non-convex constraints, the Dispatcher generates simple candidate solutions for promising, convex areas and the Selector evaluates the complicated, non-linear cost function only for the feasible candidates.. Similar to [9], manoeuvre planning is formulated as a constrained, quadratic optimal control problem, which minimizes longitudinal and lateral acceleration and jerk as well as the deviation from a reference velocity and a reference position. While there are more involved approaches from multi-objective optimization theory, currently the simple strategy of selecting according to the weighted sum of the costs is applied. If a Manoeuvre Planner is provided with a

LaneFollowingView, it applies the minimum of the appropriate distances to a currently preceding vehicle, a potentially merging vehicle and the velocity constraints (speed limit, lane curvature) for generation of position and velocity reference. The lateral trajectory is constrained by the borders of the vehicle's current lane. If a Manoeuvre Planner is provided by a LaneChangeView, the longitudinal velocity and position reference is governed by the goal to align to a certain position in the traffic gap selected by the LaneChangeView. In the lateral direction, constraints are switched from the intermediate lane border to the target lane's outer border as soon as the longitudinal profile has reached sufficient alignment to the gap. The qpOASES [10] library is applied to solve the optimization problems. The standard metric for manoeuvre selection is based on the navigation information and the acceleration effort (fuel cost) of a manoeuvre. The cost-to-go for a position at the end of the manoeuvre is queried and the manoeuvre with the minimum trade-off between cost-to-go and acceleration effort is executed.

The requirements of the first iteration tests, which are shown in D7.1 [1] and also addressed in section 3 have been realized by additions to several sub-components. The modifications and their effect are described in the following:

M1: Appropriate reaction to a notification of a road block or lane clearance by an RSU: An RSU may use a DENM message to declare individual lanes to be non-drivable. A reference geo location, a blocked distance interval and a bit-array indicating the state of individual lanes are provided by the message. The Map Provider component is modified to receive DENM messages. The referenced position in the DENM is matched to a lane cross section in the HD map. For that cross section, the bit-array is applied, closing lanes of the HD map in the process, updating its HD map representation of the according lanes. The Map Provider sends an update to the Tactical Planner, which removes the affected lane areas from the set of drivable lanes. The update is also sent to the Navigation component which then re-computes the cost-to-go values and sends updated cost-to-go values to the Tactical Planner component. An RSU may similarly modify the type of a lane with the help of a MAPEM message. For example, a road-block may be circumnavigated by clearing a certain lane for regular passenger vehicle traffic. When a MAPEM is received, the Map Provider matches the lanes in the MAPEM to the lanes in the HD map. Furthermore, if a drivable lane is prohibited in the HD map but permitted in the MAPEM, the lane status is changed to "permitted" in the local HD map. Similar to the DENM approach, the Map Provider component is modified to monitor MAPEM messages and to send appropriate updates to the knowledge base of the Tactical Planner and Navigation. The Tactical Planner reacts to the updates in the next planning cycle with standard behaviours. The removal of drivable lanes induces the planner to avoid entering the given area, whereas the modification of the cost-to-go changes the manoeuvre selection and induces lane changes according to the given situation.

M2: Execution of a Minimum Risk Manoeuvre as a reaction to a failed transition of control due to a blocked road or an advice from an RSU (MCM-ToC): During automated driving, a human on the driver seat is not involved in the driving task. For several reasons, it can be necessary to transition back the control of the vehicle from the automation system to the human driver. In the TransAID project, two causes for a transition of control (ToC) are determined: The road operator/authority may decide to disallow automated traffic in a certain area. In this case an RSU can be employed to send MCM-ToC messages to individual vehicles. Another cause is the limitation of the automated driving function: If the current goal becomes unattainable, the vehicle has to yield back control to the human driver. An orderly transition of control requires sufficient time for the human to regain situation awareness and physically take back the control. Therefore, each transition of control consists of three phases with the time intervals $[t_0, t_1]$, $[t_1, t_2]$ and $[t_2, t_3]$: Between t_0 and t_1 the driver is notified that a transition of control will have to be executed in the near future. Between t_1 and t_2 the driver is notified that she has to regain control in the next $t_2 - t$

seconds. If the driver has not taken over control until t_2, a so-called Minimum Risk Manoeuvre (MRM) is automatically executed between t_2 and t_3.

The vehicle has to automatically reach a safe state and standstill until t_3. (During $[t_2, t_3]$ the driver may take over the control and thereby cancel the automatic execution of the MRM.) An MRM is defined as a manoeuvre, which uses zero velocity and the corresponding position profile as a reference for its optimization problem, in order to stop the vehicle as fast as possible while maintaining a certain acceleration bound. The acceleration bound is chosen as the usual, minimum acceleration for *nominal* automated operation. The MRM should be distinguished from emergency manoeuvres with full deceleration capability: During an MRM and in contrast to an emergency manoeuvre, the vehicle automation system is still fully operational and starts in an uncritical traffic situation. An abrupt deceleration could negatively impact the safety of the traffic situation and would not minimize the overall risk. (If the situation should deteriorate, an MRM can still be replaced by an emergency manoeuvre with full deceleration capability.) The Dispatcher sub-component of the Tactical Planner is augmented to request planning of three additional manoeuvres: Lane following for the current lane and lane changes to both adjacent lanes, each manoeuvre with the objective of speed minimization, here denoted MRM. The Selector sub-component is modified to select only from the MRMs, if a ToC is active and in phase three, $t \in [t_2, t_3]$.

The Tactical Planner component is modified to receive MCM-ToC messages. The MCM-ToC message specifies a start position, an end position and a trigger time. Presumably, the three fields indicate the precise timing of the three phases of the ToC. It should be taken into account though, that important arguments can be made against an over-specification of the realization of such a ToC manoeuvre: First of all, the responsibility of an orderly transition of control is expected to lie with the vehicle, the automation system and the vehicle’s manufacturer. Therefore, the AV should probably decide the timing and duration of the phases on its own. Furthermore, it is inconvenient to start the MRM at a predetermined position or time, if it has to end at a fixed position, e.g. the start of the No-AD-Zone. It was therefore determined to comply to the end position (start of the No-AD-Zone) only. The points of time t_1, t_2 and t_3 are computed backwards from the end position defined by the MCM-ToC message, using minimum acceleration allowed for nominal operation during $[t_2, t_3]$ and the currently executed speed profile during $[t_0, t_2]$.

The second cause for the triggering of a ToC, an unattainable goal, is detected with the help of the Navigation component. If the minimum attainable cost-to-go is infinite, a ToC (including an MRM if driver is not responding) is scheduled. If a road-block is detected inside the sensor range, it is used as the end point of the ToC/MRM. The remaining procedure equals the procedure for a message triggered MRM described above.

M3: Changing lanes based on an advice by an RSU (MCM-LA): An RSU may influence the merging behaviour of a CAV with the help of an MCM Lane Advice (MCM-LA), possibly selecting a merging strategy for multiple vehicles in a certain area, which is optimal for traffic flow. An MCM-LA message (see Annexes A1/B1) specifies the target lane ID, the station ID of a vehicle in front of a targeted gap, the station ID of a vehicle currently following the targeted gap, a lane change start position and a start time. Both station IDs and the start constraints are optional fields. The transmission of a single station ID is sufficient to uniquely identify a certain gap. No available station IDs is in the first project iteration interpreted as an advice to change into any gap of the target lane, with the gap selection strategy at the discretion of the recipient CAV. This is currently not in line with the definitions done so far, where the automation may stick to the lane change position and timing which is provided. If either the position or time constraint are unspecified, the according dimension is here interpreted to be unconstrained.

In order to model a proper reaction of the CAV to an MCM-LA reception, the Tactical Planner component is modified to receive the message. Furthermore, the Dispatcher sub-component is modified to pose the manoeuvre planning problems in such a manner that suitable trajectories are computed: If an LA with at least one valid station ID exists, the GapRatingView discounts all gaps with a constant cost offset, where either the leading or the following vehicle match the according station IDs. Lane change planners are parametrized to plan for the minimum cost gap, taking the discount into account. If the LA specifies constraints, these are added to the constraints of the lane change planning problems. In the Selector sub-component a penalty for the discrepancy between advised lane ID of the MCM-LA i_{LA} and the goal point lane ID of a manoeuvre i_{MG} is introduced. With a penalty factor k_{LA}, the additional cost term $c_{LA} \coloneqq k_{LA} \cdot |i_{LA} - i_{MG}|$ is considered for manoeuvre selection. Evidently, this strategy allows the CAV to execute multiple, consecutive lane changes to reach the advised lane.

M4: Executing a Minimum Risk Manoeuvre into an assigned Safe Spot (MCM-ToC, MCM-LA, MAPEM): Using the MCM TransitionOfControl container, an RSU may set up a “No-AD” zone, with a transition area before it. Inevitably, a certain amount of drivers will fail to re-gain control of their CAV, leading to the execution of minimum risk manoeuvres. In such a situation, CAVs should not stop on a driving lane in order to avoid impacting the traffic flow. In many highway scenarios an emergency lane exists and CAVs could independently decide to finish MRMs on such an emergency lane. A possible strategy would be for the CAVs to queue up on the emergency lane, closing ranks at low speed if preceding vehicles exit the emergency lane, in order to clear the upstream part of the lane for further MRMs. In urban scenarios, discrete parking boxes might replace an emergency lane. An RSU may monitor occupancy of the parking boxes and advice CAVs which box to use in case of a failed transition. Using a MAPEM, parking boxes may be declared as lanes of type “park”. An MCM Lane Advice may be used to direct the vehicle onto a parking lane.

M5: Changing speed based on an advice by an RSU (MCM-SA): In order to influence the speed of a CAV, an RSU may send an MCM message with a CarFollowingAdvice (CFA) container. The message field “desiredBehavior” either contains a “TargetSpeed” or a “TargetGap”. Further, the message specifies an “advicePosition” and an “advicedLaneID”. The “advicePosition” indicates at which distance along the road the advice becomes active. It should be noted that the duration of validity of the advice is not specifically upper bounded. Presumably, the speed advice ends, when the lane with given ID ends. The Tactical Planner component is modified to receive MCM-CFA messages. On reception of a “TargetSpeed”, the speed-limit of each manoeuvre planner instance is upper-bounded by the specified value.

M6: Opening a gap based on an advice by an RSU (MCM-SA): As discussed in M5, an MCM’s CarFollowingAdvice container may specify a “TargetGap”. Supposedly, the target gap size should be sent from an RSU to a CAV to support the merging of another vehicle in front of the CAV. Unfortunately, the specification heavily depends on the uninvolved vehicle initially in front of the CAV in the same lane. If the uninvolved vehicle does not exist, the gap size is undefined. If the uninvolved vehicle accelerates or “disappears” (by changing lanes), the CAV has no viable reference upon which to support the merging manoeuvre. Therefore, the value of “TargetGap” is interpreted as minimum target gap. If the vehicle in front is not available, compliance to the distance value is given. The ID of the merging vehicle is not transmitted in the current format. If a vehicle enters the lane in front of the CAV, the CAV cannot determine whether this was the intended vehicle or whether the gap still has to be maintained. It is therefore recommended to modify the message by which an RSU may request a CAV to support a merging manoeuvre: A simple solution could be to specify the station ID of the merging vehicle and to broadcast CPM messages containing state information of the merging vehicle. In this way, the CAV is enabled to continuously and foresightedly adapt its speed to support the merging process. This approach would also solve the issues with M5.

M7: Sending and receiving planned manoeuvres via MCM (MCM-VMC): The Tactical Planner component is modified to send and receive MCM with a "VehicleManoeuvreContainer" (MCM-VMC). Each time a trajectory is selected for execution, an MCM-VMC is sent. On reception of an MCM-VMC, it is evaluated, whether the planned manoeuvre is useful for traffic prediction. A filter is applied, which determines, whether the sending vehicle is relevant and whether it has precedence over the ego vehicle. Irrelevant plans are discarded, relevant plans are maintained in a set C_p for a limited amount of time (or until they are replaced with a new message originating from the same station ID).

M8: Detecting the necessity of cooperation and broadcasting a desired manoeuvre via MCM (MCM- VMC): The set C_p is applied for prediction of traffic participants. Predictions are used for the specification of constraints for the manoeuvre planners. To determine that the ego vehicle requires cooperation for a specific manoeuvre it is insufficient to know that a certain manoeuvre is infeasible under the current set of constraints/predictions. Additionally, the knowledge is required that the modification of the behaviour of another traffic participant enables the feasibility of a certain manoeuvre, or that it reduces its cost. To acquire that knowledge, the Dispatcher sub-component is modified to request planning of an additional, "hypothetical" manoeuvre: In this manoeuvre, the prediction of one or more traffic participants is replaced by a "hypothetical" cooperation behaviour. Such a manoeuvre is never selectable for execution and merely serves to compare cost and feasibility. If the necessity of cooperation is thus determined, the "hypothetical" manoeuvre is added to the MCM-VMC container as a desired manoeuvre.

M9: Determining an appropriate reaction to the reception of an MCM desired manoeuvre (MCM-V2V): The reception of desired manoeuvres is handled similarly to the reception of planned manoeuvres described in M7. In addition, the difference in required acceleration effort is estimated and only desired manoeuvres below a certain threshold are added to C_p. If a desired manoeuvre is added to C_p, the ego vehicle's affected plans are "automatically" adapted to support the cooperation request. In that case, the own planned trajectory is updated in the MCM, allowing the vehicle which was expressing its desire to follow it.

2.1.1.3 Communication

The V2X communication module is logically divided into the V2X message creator and the V2X radio interface modules. The V2X radio interface is implemented in TransAID at the DLR prototypes by using the Cohda's MK5 On-board Unit (OBU), while the V2X message creator runs in the Car-PC where the Dominion Framework is installed. A wired Ethernet connection enables the communication between the V2X radio interface (i.e. Cohda's MK5 OBU) and the V2X message creator (i.e. Car-PC). Figure 4 shows the existing interfaces between the two modules.

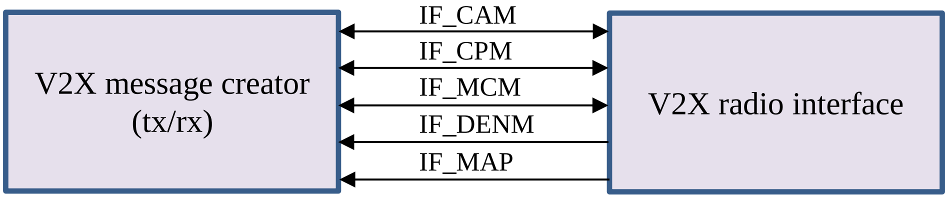


Figure 4: Architecture of the V2X communication module at the vehicle

2.1.1.3.1 V2X message creator

The V2X message creator module serves as a *middleware* to facilitate the integration between the Dominion software and the software running on the V2X radio interface. TransAID has followed this modular design approach to minimize the impact of substituting or evolving any of the two software the V2X message creator is connected to, and to facilitate the independent development of the different blocks. The communication between the dominion and the V2X message creator, and between the V2X message creator and the V2X radio interface, is enabled through UDP sockets.

The architecture of the V2X message creator module is represented in Figure 5 and Figure 6 from the transmission and reception point of view, respectively. At the transmission side (Figure 5), the information generated at the Dominion Framework and transmitted through the interfaces 1a, 2a and 4 (see Section 2.1 in D7.1), is received at the UDP sockets and used to populate the CAM, CPM and MCM messages. Then, those messages are transmitted through other UDP sockets towards the V2X radio interface module. On the other hand, at the reception side (Figure 6) the V2X message creator module receives the content of the CAM, CPM, MCM, DENM and MAP messages through different interfaces, and after depopulating them, their content is transmitted through the interfaces 1b, 2b, 3, 5 towards the Dominion Framework (see Section 2.1 in D7.1). Both in the transmission and reception sides, some transformations of the messages' data are required in order to adapt them to the V2X radio interface and Dominion Framework requirements.

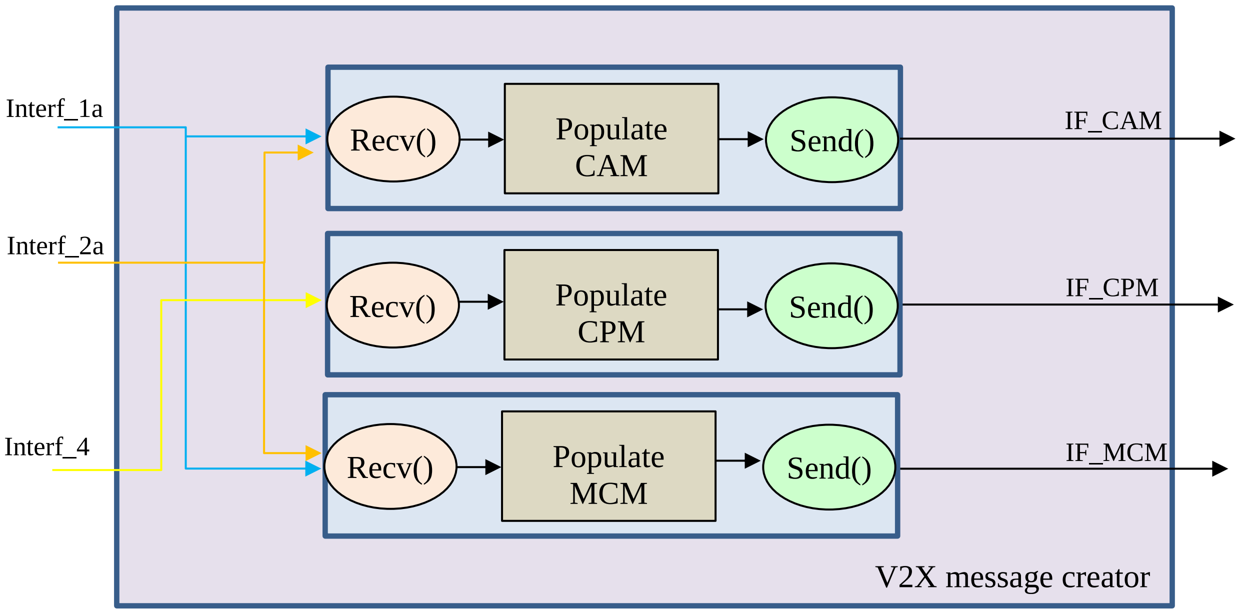


Figure 5: V2X message creator: transmission

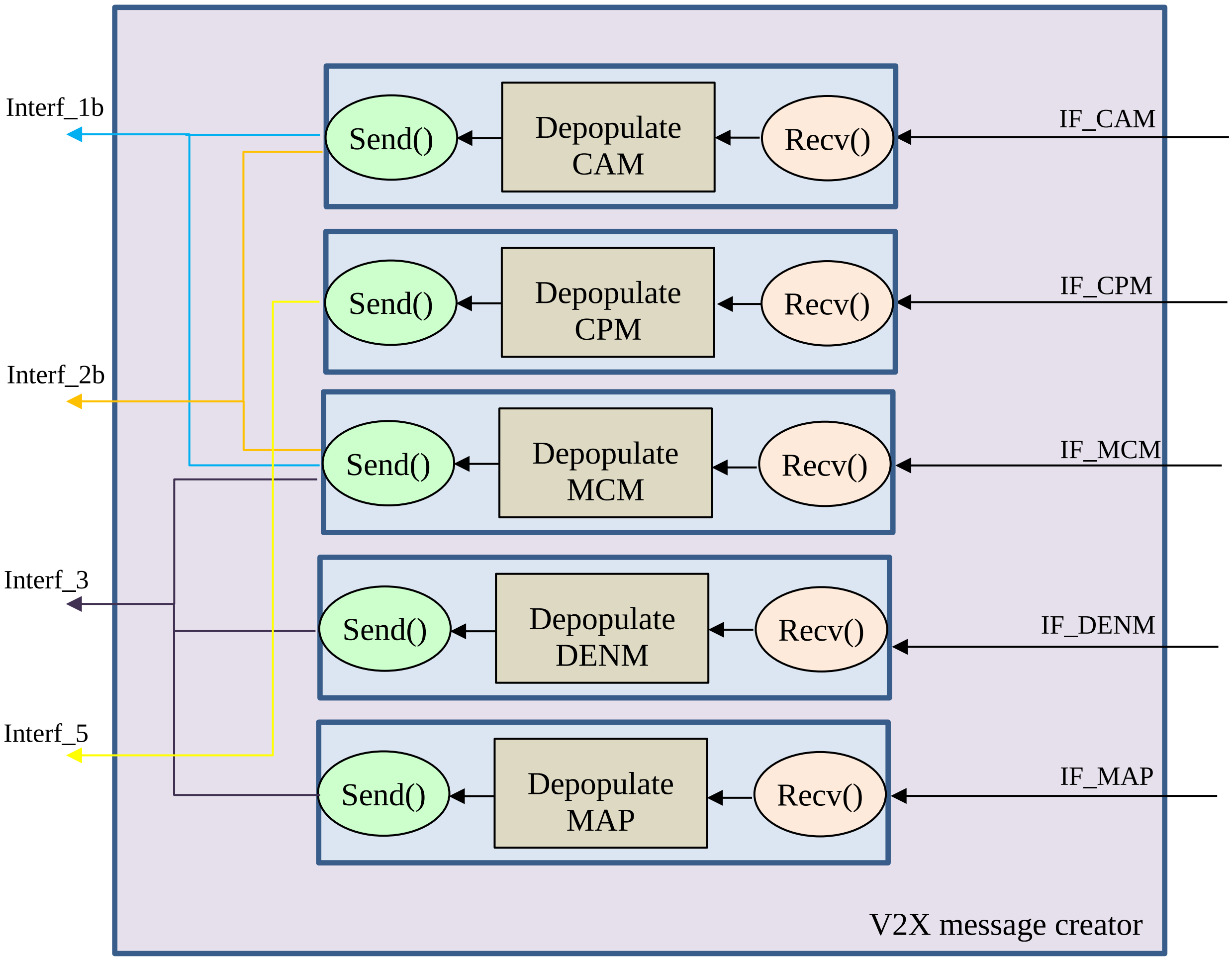


Figure 6: V2X message creator: reception

2.1.1.3.2 V2X radio interface

V2X communications in TransAID are enabled by the use of commercially available off-the-shelf (COTS) ETSI ITS G5 solutions compatible with the latest stable versions of the ETSI ITS and SAE DSRC standards [11] [12] [13] [14] [15] [16] [17] [18] [19] [20] [21]. TransAID has implemented its communication protocols and message sets on the top of these solutions thanks to the extensibility properties offered by them. In particular, for the case of DLR the Cohda's MK5 OBU has been selected which includes a Software Development Kit (SDK) that enables and facilitates modifications and customizations of the ETSI ITS G5. The resulting TransAID V2X radio interface module architecture is depicted in Figure 7. As it can be seen, a TransAID radio interface module is compliant to the standard ETSI ITS communication architecture [12] and supports transmission and reception of V2X messages over the ETSI ITS G5 radio technology as profiled in [20]. The adopted network and transport layer protocols are exactly the same as standardized in [15] - [19] and implemented in the commercially available V2X solutions, which provides a straightforward approach to bring TransAID implementations on real-road tests. In addition, the TransAID V2X radio interface module implements the Facility Layer's functional requirements and specifications as described in [15] including the support for DENM and CAM basic services, and the maintenance of the Local Dynamic Map and Vehicle State databases. On top of this, TransAID has extended the ITS G5 Applications to accommodate the needs of the TransAID use cases/services. Several V2X services have been created from scratch to manage the transmission and reception of MAP, CAM and DENM messages (extending the Decentralized Environmental Notification and Cooperative Awareness services), and CPM and MCM messages to enable the Collective Perception and Maneuver Coordination services, respectively. This is represented in Figure 7 by the Application Layer's CAM, MCM, CPM, DENM and MAP modules. These modules implement the functionalities to manage V2X messages to be transmitted and/or received, including UPER co/decoding and information processing.

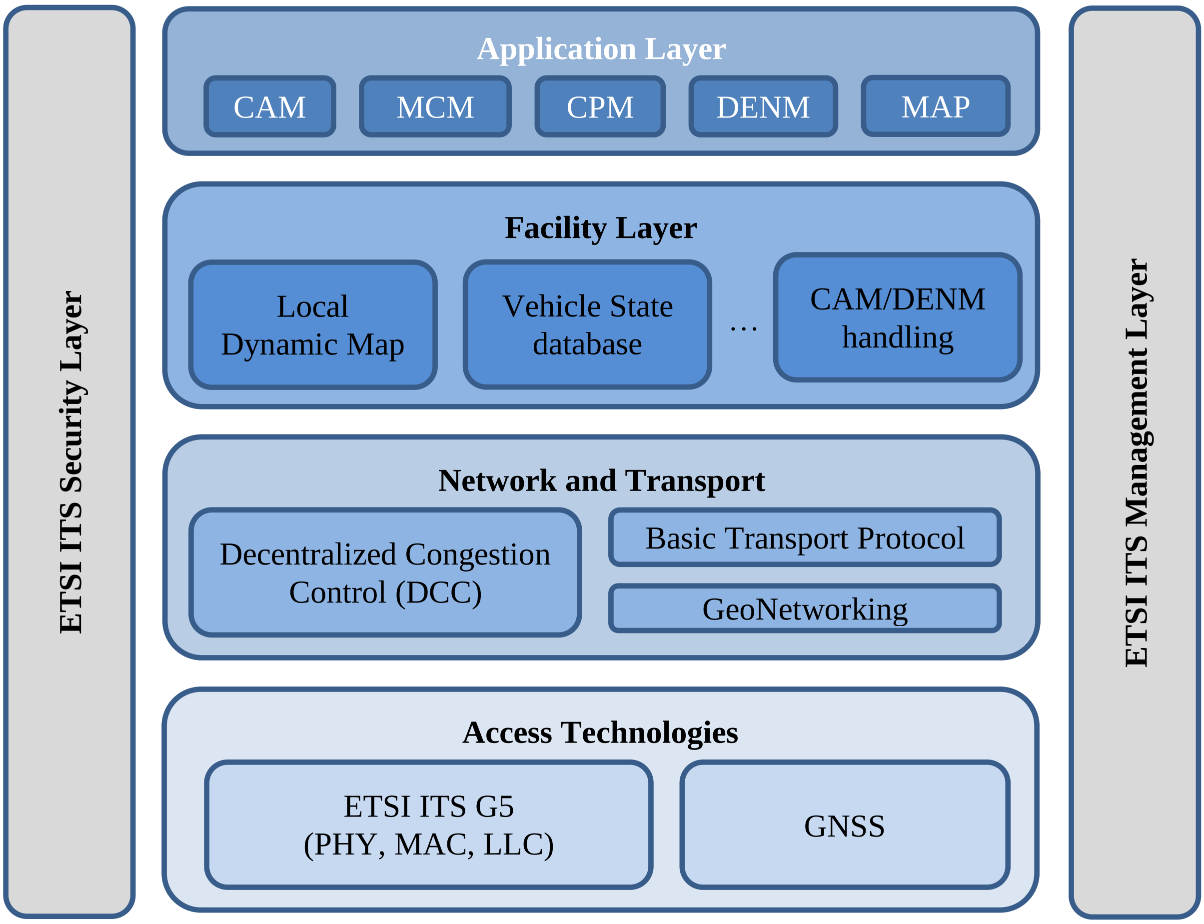
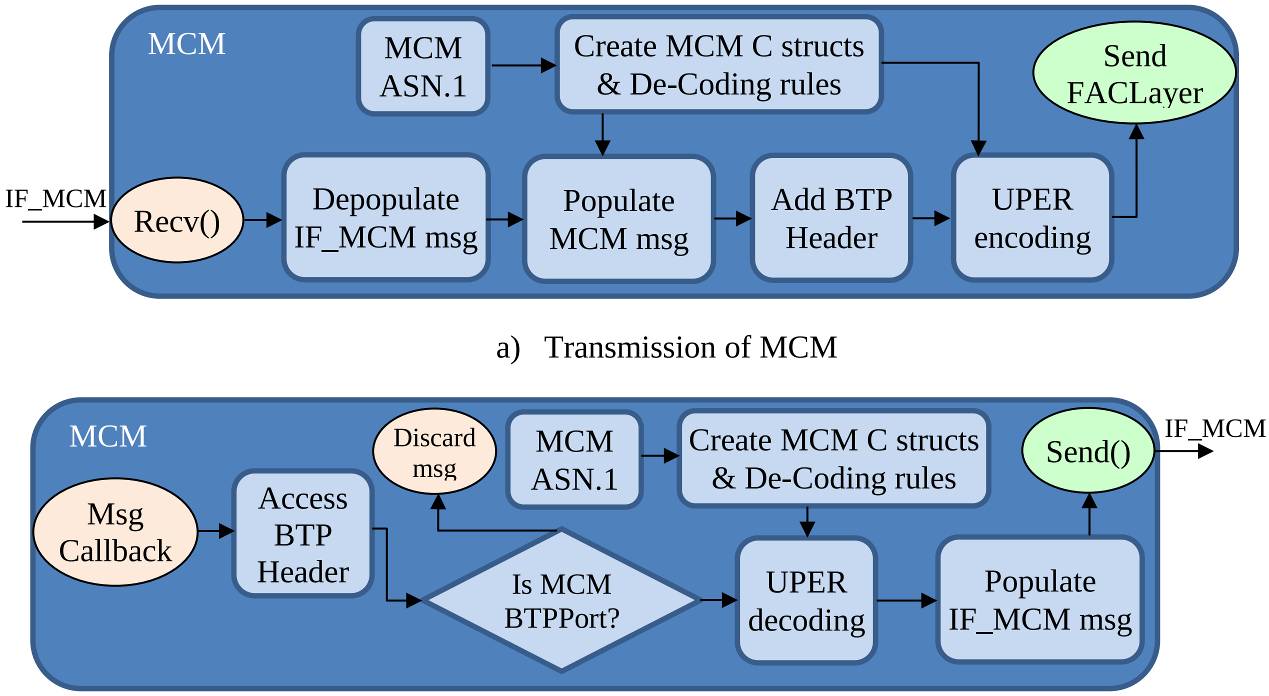


Figure 7: TransAID V2X radio interface architecture

Using as an example the MCM module of the ITS G5 Application layer depicted in Figure 7, Figure 8 shows the processing of MCM messages on the transmission (see Figure 7.a) and reception (see Figure 7.b) path. On the transmission path, the MCM application takes the information of the IF_MCM interface coming from the V2X message creator module and depopulates it. Then, using the C structs that are created out of the MCM ASN.1 definition, the MCM message is populated and the BTP header is added. It is important to note that integrating the MCM ASN.1 definition in the MCM application module provides high flexibility to modify and adapt the message's container to the TransAID requirements. The MCM ASN.1 definition is also used to generate the coding rules for the MCM's UPER encoding. The resulting message is then transmitted to the Facility Layer. The MCM ANS.1 definition used during the first iteration of TransAID is included in "Annex B1: MCM ASN.1 specification" (the description of the different fields is in "Annex A1: MCM description"). On the reception path, messages arrive at the Application layer through a callback function. All TransAID Application Layer's modules use a similar callback function that is invoked when any of these messages is received. Therefore, the information of the BTP header needs to be accessed in order to identify the messages that are to be processed in this module, e.g. MCM. For example, for the MCM message TransAID has set the BTP port 2010, while CAM and DENM messages are identified by the BTP port 2002 and 2001, respectively. The MCM ASN.1 definition is also used to create the UPER decoding rules that are used to get the information of the MCM message, and finally to populate the interface message to be transmitted through IF_MCM.



a) Transmission of MCM

b) Reception of MCM

Figure 8: TransAID V2X radio interface architecture: application layer a) transmission, b) reception (MCM used as an example; a similar approach used for other applications such as CPM, CAM, DENM, and MAP)

2.1.1.4 Debugging HMI

Although TransAID does not deal with HMI in general, it has been decided to implement a debugging HMI for testing and also for demonstration of the behaviour. The HMI is not fulfilling current state-of-the-art HMI paradigms and is only for displaying the internals of the vehicle automation's decisions and respected inputs.

Figure 9: Debugging HMI overview

As shown in Figure 9, the HMI consists of standard elements like the revolution counter and the speedometer. The additional center part element consists of the following elements:

- Text Box: Here, additional text is shown.
- Speed Advice: Whenever a speed advice is received via MCM, it is directly shown here, converted to km/h.
- Lane Change Advice: Whenever a lane change advice is received via MCM, it is directly shown here. The Lane Change Advice consists of a couple of values: On the left, the current and desired lane ID is shown in the format "current > desired". On the right side, the distance to the lane change position is shown. The arrow indicating the lane change direction is either turning left or right. Furthermore, it is either pulsing in case of a pending lane change or solid in case it is currently executed.
- Transition of Control Advice: This field is composed of a hatched area and the remaining distance in the current driving mode. The hatched area can be either yellow in case of a transition of control taking place or red in case of a minimum risk maneuver. The area is either pulsing when the advice is pending or solid when the advice is active. In case a minimum risk maneuver is executed, the hatched area is replaced by a warning message box, shown in Figure 10. A Transition of Control Advice is accompanied by the text message "Take Over Control!" shown in the Text Box. Therefore, a normal transition from automated control to human control without any action to take over by the driver consists of the following steps:
    1. Active transition of control: pulsing yellow hatched area and distance value
    2. Active minimum risk maneuver with overlay image.

All other combinations may only occur in case of wrongly used values. Being a debugging HMI, these cases nevertheless may occur.

Figure 10: HMI showing active Minimum Risk Maneuver

2.1.2 CVs

In the first project iteration, only one single CV has been used, but only for testing the sensor data fusion in the CAVs of Lidar and CAM as described earlier.

The used car was a Volkswagen T5 bus, which used a Cohda V2X Box for communication. In addition, the bus includes an inertial measurement unit (IMU) coupled with a high-precision satellite navigation systems similar to those in the used CAVs. Therefore, the positioning data included in the CAMs was of high precision.

Figure 11: DLR's T5 bus used as a CV in TransAID

2.2 Road Side

All tests in the first project iteration have been performed on the Peine-Eddesse test track already described in D7.1. In this iteration, a virtual road topology (Figure 12) has been placed on the test track which consists of a two-lane straight road which can be used as highway or rural road. It is accompanied by a merging lane used for the merging Service 2.1 (see D7.1 for details).

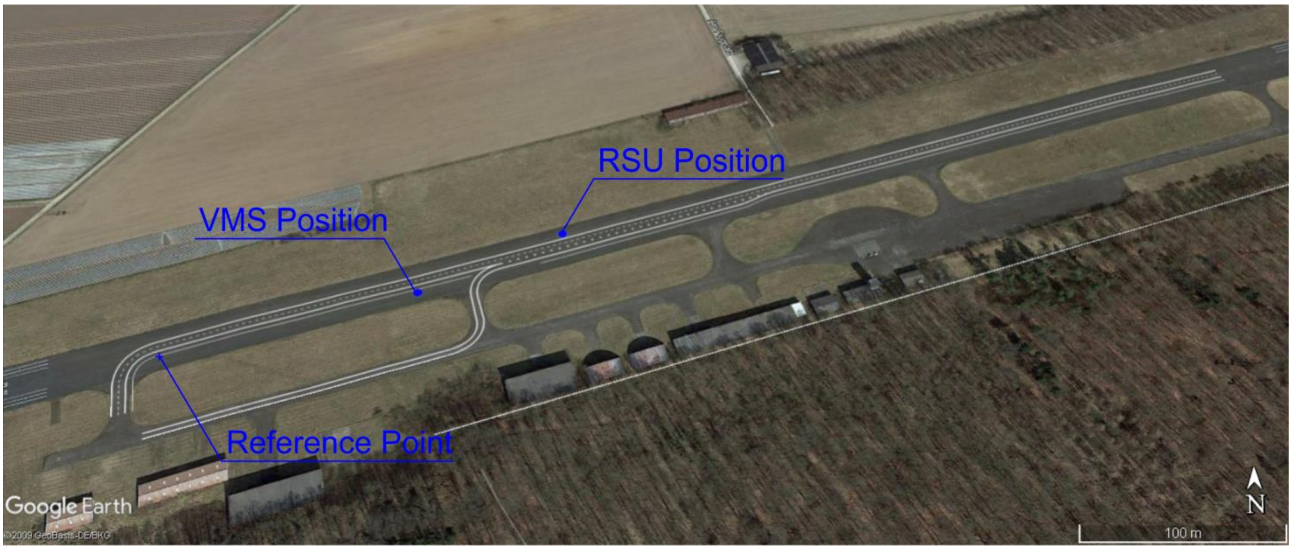


Figure 12: Used road topology on the Peine-Eddesse test track

In addition to the virtual parts, a variable message sign and a RSU and Camera pole has been placed on the test track at the indicated positions.

Furthermore, a reference point was included in the first iteration trials. This reference point is used as local reference of lane, speed or ToC advice positions instead of using the content of a MAPEM, which will be used in future implementations.

In the following, all parts are explained in detail.

2.2.1.1 Sensors and Sensor Fusion

The mobile RSU used at the test track on Peine-Eddesse Air Field consists of a mobile retractable pole with an ACTi camera type B94 mounted at the top. This outdoor camera has a maximum resolution of 1.3 Megapixel and can record videos at 30 fps with a resolution of 1280x960 pixels. Furthermore it is contained in a weatherproof casing and is equipped with a fan and heater that are like the camera powered by Power over Ethernet (PoE). The recorded data is processed on an ECX-1200 computer with an integrated NVIDIA GeForce GTX 1050 graphics card to allow for fast inference time of the subsequent object detection. The described setup is depicted in Figure 13.

Figure 13: Mobile RSU with mounted ACTi Camera and ECX 1200 processing computer

Since the demonstration use cases within TransAID aim to leverage synergies between the infrastructure and automated vehicles, the videos recorded by the camera are further processed with the aim of providing relevant object information to the passing vehicles. This pipeline for further processing the recorded videos is implemented in ROS. Therefore the video stream from the camera is first read into the format of a ROS message, before being passed to a node performing object detection. The object detection is performed by a neural network. Specifically a tensorflow implementation of a ResNet-50 network architecture comprising a Faster-R-CNN as detection algorithm is used. The network was trained on a manually labelled dataset acquired at the DLR reference track and is able to detect and classify cars, vans, trucks and their trailers as well as busses, motorbikes, pedestrians and bicycles. The detected objects are subsequently tracked over time in order to determine object velocities, reduce uncertainties and also provide object histories. For this an adapted version of the approach presented in [22] is implemented. The tracking is based on a Kalman filter that performs the prediction step based on a constant velocity model. The predicted tracks are matched to the new detections with linear assignment based on a cost matrix. In doing so, confirmed tracks of objects that have already been tracked over multiple time steps are associated first, further processing consistently tracked objects prior to tracks with gaps in their tracking history. Matched tracks and detections are used to update the Kalman filter while unmatched detections generate new track candidates. The tracked bounding boxes in image coordinates are then transformed into the UTM coordinate system based on the calibrated inner and outer orientation and the known position of the camera. For the succeeding V2X message transfer, the data is formatted into a ROS message in the CPM format, ensuring the correct value ranges and units and handling invalid entries. In the final step these ROS messages are converted to UDP packets that are sent to a java application for further communication to the Cohda V2X box also mounted on the mobile RSU.

2.2.1.2 Traffic Management System

The design of the traffic management system was scenario-driven at this stage of the tests. There is a receiver for CAMs from connected vehicles and there are senders for MCM, MAP and DENM running on the RSU. Depending on the scenario, each of the outgoing messages was either enabled or disabled (see Section 3 for more detailed information).

Each scenario was defined by a .conf file and a Java script containing the traffic management logic. In the .conf file, one can enable or disable the sending of specific messages, define the output ports for each message type and define the identifiers of the vehicles involved in the scenario. The Java scripts ran specific instructions for the respective scenario, i.e. static messages were sent. Future works for the 2nd iteration on the traffic management system will include the generation of dynamic messages based on CAM data, CPM data for safe spot availability, and different acknowledgements for ToC/MRM performances, safe spot assignments or automation mode in order to emulate the services provided by the other work packages.

2.2.1.3 Communication

The design of the V2X communications module at the infrastructure is similar to the one implemented at the CAV (see Section 2.1.1). The Cohda solution used in this case is the MK5 RSU which is built with the same chipset as the MK5 OBU used in the vehicle (see Section 2.1.1.3), but housed in a waterproof enclosure. The DLR MK5 RSU solutions are also Power over Ethernet (PoE) capable.

In this case, the V2X message creator module is logically divided into the V2X message receiver and V2X message sender as depicted in **Error! Reference source not found.**. Besides, the onfiguration files used for the V2X radio interface allow indicating whether the ETSI ITS G5 V2X solution should act as a passenger vehicle, or as an RSU.

2.2.1.4 Variable Message Sign

As variable message sign a Niechoj electronics LUMEX full matrix sign compliant with EN 12966 has been used. For displaying, this device is receiving full colour bitmap files in the resolution of 39 x 40 pixels via Ethernet. During the integration of the first project iteration, an application has been developed which is updating the shown images frequently, according to the needs of the shown scenario. This approach also allows changing images or animations.

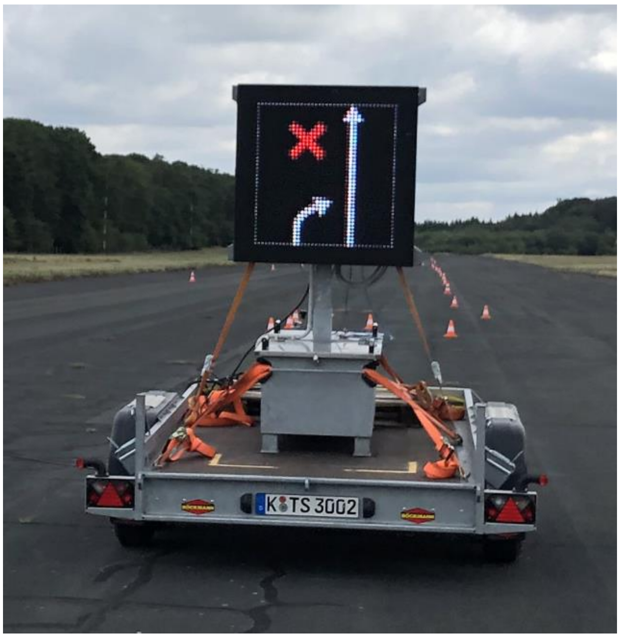


Figure 14: Variable message sign used during the test runs of the first iteration

3 Feasibility assessment

This chapter describes the general setup of the feasibility assessment.

3.1 General procedure

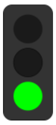
In general, the feasibility assessment is prepared by the WP7 partners. While automated vehicles are prepared by DLR only, the road side equipment is prepared by DLR in the first iteration and by DLR and Dynniq in the second iteration. The communication aspects are developed by UMH. Details of all used parts have been described in chapter 2.

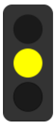
Requirements which need to be fulfilled by the prototype (vehicle and infrastructure) have been proclaimed in D7.1. Basically, there are general requirements and requirements per scenario.

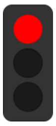
After preparation, HMETC is visiting the test tracks and testing the prototype in the different scenarios. The feasibility assessment itself consists of

a) Requirements verification
b) User experience
c) Summary of the overall feasibility

The requirements verification is done by rating the successfulness of each of requirement. Therefore, each requirement is referenced from D7.1, rated and annotated. The rating follows this scheme:


The requirement is completely fulfilled.


The requirement is partially fulfilled. Details are given in the annotations.


The requirement is not fulfilled. Details are given in the annotations.

All aspects of the feasibility assessment are shown in the following.

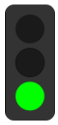
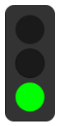
3.2 First iteration

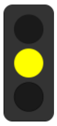
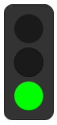
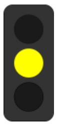
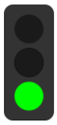
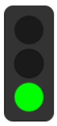
During the first project iteration, a set of scenarios had to be tested. The scenarios are introduced in D2.2 [2] and further specified in D7.1 in terms of real-world assessment.

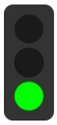
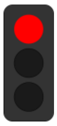
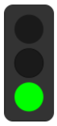
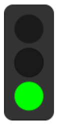
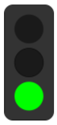
In the following, the feasibility assessment of the first iteration is shown. After dealing with the general requirements, the specific requirements for the first iteration scenarios are described.

3.2.1 General requirements assessment

3.2.1.1 Requirements verification

General requirement description		Req. Name	Associated Test cases successfully executed	Notes
Vehicle requirements	*Availability of cooperative automated vehicles:* As TransAID deals with transition areas, all scenarios include at least one cooperative automated vehicle. Therefore, cooperative automated vehicles need to be available for the feasibility assessment. The vehicles need to be able to drive longitudinally and laterally automated, independent of the SAE level of automation, as well as to cooperate via V2X.	REQ_V_G_1		The minimum required number of CAVs was present during the tests
	Availability of transitions of control As TransAID focusses on SAE levels up to level 4, the automated vehicles need to have the ability to perform transitions of control to the driver and from the driver to the vehicle automation. The transitions need to be driver and automation initiated, meaning that the driver may decide which system is turned on (for each longitudinal and lateral control either manual driving with warnings or automated driving), but the automation itself may decide to not being able to keep the desired level of automation any longer.	REQ_V_G_2		Transitions of control could be executed
	Availability of Minimum Risk Manoeuvres (MRM) Whenever the automation is not able to continue driving at the desired level of automation, it has to try to give the control back to someone else, most likely (in SAE up to level 4) the driver of the car and sometimes a remote operator. Whenever this take-over-request (ToR) is not followed by the driver due to any reason (very distracted, fallen asleep, lost consciousness), the SAE4 vehicle has to reach a safe state. This is done by automatically triggering a Minimum Risk Manoeuvre. While this is especially true for SAE4 vehicles, it is foreseen that SAE3 vehicles will also offer light versions of such MRMs, e.g. decelerating to a full stop of the vehicle on the current lane. Nevertheless, current thoughts of MRMs also include lane changes to emergency lanes, and therefore more sophisticated behaviours. Vehicles driving in lower levels of automation do not have MRMs, as the driver always has to monitor the situation and as such is already in the loop. During the feasibility assessments of TransAID, Minimum Risk Manoeuvres need to be available in different kinds, so that different SAE levels can be tested.	REQ_V_G_3		Standard and extended Minimum Risk Maneuvers could be executed.

	Availability of extensible sensor data fusion The automated vehicles will need a sensor data fusion, which will fuse the data of the different sensors. This will need to be extensible, as it is foreseen that further data will be added to it, e.g. data related to map properties (availability of safe spots, see Scenario 4.2), or data received by cooperative perception. The latter will include data from other vehicles' sensors or from infrastructure sensors.	REQ_V_G_4		The sensor data fusion is available and has included interfaces for CPM and CAM perception. Only CAM-Lidar fusion is currently used. In addition, map properties are changed according to DENM and MAPEM receptions. Nevertheless, the fusion with the CPM objects has not been implemented yet.
	Communication and message sets As TransAID is relying on V2X communication based on the ETSI ITS-G5 radio access technology and its associated ETSI ITS standards, each cooperative vehicle has to be equipped with the appropriate hard- and software to receive and send dedicated messages on the given channels.	REQ_V_G_5		Communication is implemented following the designed message sets.
	Cooperative lane changes One of the key abilities repeated in several scenarios is the ability to perform cooperative lane changes. While the precise communication for such cooperative lane changes is going to be studied in WP5, it is nevertheless a basic requirement for all cooperative automated vehicles to be able to perform cooperative lane changes.	REQ_V_G_6		Cooperative lane changes in terms of V2V cooperation has only been tested in simulation, see sections 3.2.3.2.2 and 3.2.3.2.3
	Local high definition map The automated vehicles need to have a local high definition map of the scenario area. This map needs to include a detailed representation of the road topology as required by automated vehicles implementations, and must be extensible to include additional dynamic data sent by the infrastructure, like road works areas, positions of safe spots etc.	REQ_V_G_7		A local high definition map was present. As mentioned in REQ_V_G_4, the map data is already dynamically changed on reception of DENM and MAPEM.
	HMI availability for CVs Task 5.5 describes signalling for legacy and cooperative vehicles, including signalling inside the vehicle. For this, the vehicle needs to have an HMI available. This will most likely be an Android smartphone connected to the OBU.	REQ_V_G_8		As no CVs were present during the tests, also the CV HMI was not needed. Instead, a debugging HMI was used in the CAVs
Infrastructure requirements	*Communication and message sets* It is a mandatory requirement for the infrastructure to be able to communicate advice to the vehicles by using ETSI ITS-G5 based V2X communication. In addition, the reception of messages is also needed to get a better image of the situation, e.g. by knowing the exact positions of cooperative vehicles and their plans, as well as knowledge of other non-cooperative vehicles' presence. To avoid extensive forwarding of messages, different road side units shall be linked to each other. While this is a general requirement, it will not be used during the feasibility assessment, as there will always be only one single road side unit available. Furthermore, the infrastructure needs the ability to communicate decisions to non-cooperative vehicles as well. This can be done by for instance Variable Message Signs. Possible additional methods are to be developed within WP4 and	REQ_I_G_1		.The infrastructure was able to communicate messages in line with the defined message sets. As mentioned in the requirements, only one single RSU has been used. Communication to non-cooperative vehicles has been done by using a VMS, see REQ_I_G_6.

	WP5.			
	Sensors In most cases, the infrastructure also needs to know where all non-cooperative vehicles are. Therefore, sensors to detect vehicle positions are a mandatory requirement. While the sensor can be of any kind, cameras are foreseen to be the best option, as they offer not only vehicle positions, but also more details, like the orientation and speed.	REQ_I_G_2		A camera was able to detect and track objects.
	Sensor data fusion As for the vehicles, also the infrastructure needs to perform a sensor data fusion, e.g. to understand that a vehicle detected by a camera is also transmitting messages.	REQ_I_G_3		In the first iteration, no sensor data fusion was present during the tests. This only affects Test 4.2_2, as in all other services no link between objects and message generation is required. Simulations for use case 2.1 have already demonstrated data fusion between sensors, CAM and CPM data. However, this will only be tested in the field in the second iteration.
	Processing capabilities The infrastructure needs to be able to compute several inputs to generate correct traffic management measures. Therefore, the infrastructure needs to include adequate processing capabilities. If the sensors need further processing capabilities e.g. to calculate object positions and dimensions, this needs to be included as well.	REQ_I_G_4		Processing was possible without any shortcomings.
	Road networks The different scenarios will need different road network topologies to be taken into account. The road networks need to be available logically so that the infrastructure is able to plan on top of it.	REQ_I_G_5		The used road network was included in the infrastructure as well.
	Signalling equipment The only method to reach non-cooperative legacy vehicles is through roadside equipment. Task 5.5 will investigate this further, but as there is no budget foreseen for Variable Message Signs (VMS), it is likely that this will be limited to existing infrastructure, e.g. traffic light signals, ramp meters, etc.	REQ_I_G_6		A VMS was available and used.

3.2.1.2 Deviations to the final implementations planned in the second project iteration

Some deviations are existing by design in the first iteration. These are summarized in the following and will be implemented during the second project iteration:

- *Surrounding Traffic:* All tests have been performed with the minimum number of required participants in order to focus on the service implementations. Therefore, no LVs or additional CVs have been used during the trials.

- *Reference Position*: The TransAID message set includes several positions of actions in the MCM triggered by the Road Side. These positions are modelled as one-dimensional integers (see Annexes A1 and B1) referring to road segments identified in the MAPEM container. Since the MAPEM container needs an intersection with ingressing and egressing lanes, which is not present in the current road topology (see chapter 2.2) it has been decided to use a hard coded reference point in the first iteration. All distances are measured along the lane from this point.

- *Camera integration:* The camera system used for the object detection was already successfully transmitting CPMs of all detected objects on the test track. Nevertheless, the object data has neither been used in the sensor data fusion of the vehicle (see chapter 2.1.1.1) nor in the road side (see chapters 2.2.1.1 and 2.2.1.2).

- *VMS images:* The images and animations have been created in correlation with Task 5.5 of the TransAID project. Nevertheless, it has to be said that the research in this task is not yet finished. Therefore, the images are not final.

3.2.1.3 User experience

This section explains what was the general experience and feeling when applying the services in real life from a car passenger/driver perspective, in order to understand if it is something that can be sold to OEMs customers.

It is important to highlight that the DLR test-vehicles are purely an experimental platform used to test and validate technical developments and not primarily meant to address perfect user experience. As mentioned before, in the performed integration sprint and demonstration the main objective was to show primarily the cooperative interaction between an automated car and the road infrastructure as well as the automated implementation of infrastructure advice.

The test vehicle successfully drove automated and executed the required manoeuvres on the test track according to the scenarios. Being a careful reviewer as passenger in one of the back seats traveling with the test vehicle didn’t feel different from a human driver. This can be already seen as a positive result of DLRs implementation, passengers don’t feel unsafe while the car is traveling in automated mode. A successful ToC was not interrupted by a sudden change of vehicle speed or a steering jerk. The test vehicles driving behaviour resulted in a safe and comfortable ride for passengers.

- In general, the applied acceleration and deceleration values were as expected comparable to a comfortable not aggressive driving stile of a human driver
- Recognizable steering jerk while being in the curve sections before entering the ramp was noticed. This could be improved by applying slower steering angle changes and lower speed while traveling in the curved sections
- In general the MRMs were recognizable but still had a smooth deceleration. As MRMs should be one of the last counter-measures before an accident it is acceptable.
- Lateral vs longitudinal speeds
- Messages received and processed in time
- Very smooth lane following on straight paths. The steering wheel was not jittering, vibrating or shaking
- In case of a requested/required lane change, a bit smoother trajectory should be planned (if possible), in terms of a not too abrupt change of lateral speeds to support a comfortable travel (this was noticeable especially when changing the lane from the ramp to one of the straight lanes). This can have influences on the path planning; a longer planned/calculated path (smaller lateral/longitudinal changes between single steps) compared to a human driver.
- Required V2X messages were transmitted and received properly to be taken into account for the individual test cases
- A HD map with overlays/status information of blocked or ending road segments was used to execute the test cases.

From an OEM perspective, potential areas for improvements can be seen in the HMI area, the reader should be aware that the used (debug) HMI is not in scope of TransAID:

- No indication of system status: automated driving vs. manual driving. A light blue colour inside of the cluster (background or as a thick borderline) could support indication of a CAV in automated driving mode. Additionally, the transition of control should be indicated using a short display pop-up message and/or audible output (text to speech function, beep, etc.).
- One or two buttons on the steering wheel (detection of driver's grip on steering wheel) could be an additional step to acknowledge transition of control.
- No turn light indicator used before and during lane change (at least inside of the vehicle not signalized using audio and/or cluster)
- Further investigations should be done for the cases where a MRM will be executed. Either before starting the MRM the driver must be warned (vibration, audible, visual with longer warning cycles) to take back control (cf. driver state monitoring) of the vehicle to reduce the number of MRMs or after executing the MRM an emergency case strategy should be started (in case the driver is not able to react), starting with warning signals and ending with signalling that external help is required (e.g. hazard lights, horn, e-call). After executing the MRM vehicle, the engine should be stopped and all doors unlocked.
- Take over requests for drivers must be signalized much clearer (at least for first time users); a red flashing exclusion zone in cluster can be misinterpreted, starting with a light yellow fading to orange and red or a progress bar might help.
- Especially in case of lane changes, it will be more comfortable to indicate the next manoeuvre to prevent the driver from countermeasures resulting in unsafe behaviour and less comfortable travels.
- Another not yet verified solution could be the decoupling of steering and pedals while the vehicle is in automated driving mode.

3.2.1.4 Check overall feasibility

This section considers the results of the requirements verification and of the user experience and derive conclusion on overall feasibility. Also, it justifies if a given service is feasible/applicable in real-world implementation scenarios and why.

All test scenarios have been tested successfully and identified as mandatory baseline for following test scenarios. These base scenarios themselves are feasible and required for a real-world implementation (cf. L4 systems). A larger-scale test setup, using multiple CAV/CV as well as LV as mixed traffic environment, would be interesting especially when executing a MRM in order to assess the impact on traffic flow. Room for improvement is seen in the HMI area: Passengers of CAV/CV could be better informed before and while the vehicle is executing manoeuvres, resulting in a comfortable and safe travel (cf. travel sickness). This lack of information is related to the early stage of the prototype, which is not specifically designed to offer an end-customer HMI. TransAID lays down the focus in a proper function and manoeuvres implementation are not focusing HMI at this stage of the project.

Overall, the implementation looks feasible from an OEMs point of view. Some test cases will be reviewed in the second test sprint to better judge the influence/impact of other road users (especially LV) and to get an impression from the outside monitoring the scenarios. ToC use cases were properly executed and implemented in a reasonable way. Further investigations could be done to select appropriate timings or distances for handing back the control to the driver.

3.2.2 Scenario 1.1: Provide path around road works via bus lane

3.2.2.1 Description of the scenario from D2.2

In most situations where road works block the normal lanes and there is a bus lane, that lane is provided as an alternative route to circumvent the road works. Automated vehicles might not have the (appropriate) logic to determine whether such an action is tolerated in the given situation (i.e. unable to detect the situation and corresponding correct lane markings) and need to perform a ToC. Also, especially in urban situations, such markings might not always be provided (in every country). By explicitly providing a path around the road works from the road side infrastructure (RSI), CAVs can drive around the road works and maintain their automated driving (AD) mode (and thus preventing a ToC). That way, it is clear where the CAV is allowed to break the traffic rules and drive across the bus lane.

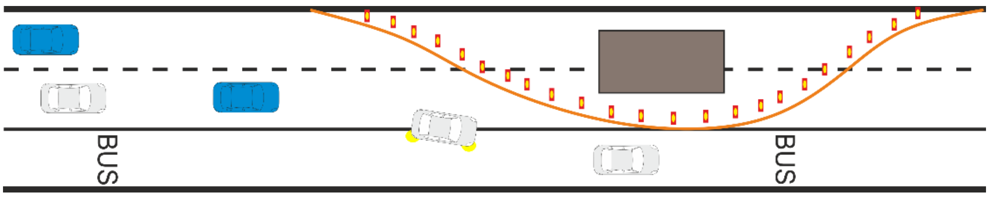


Figure 15: schematic overview of Scenario 1.1

In this scenario, there are road works on a two-lane road with a bus lane next to it. The RSI has planned a path and is distributing it. Approaching CAVs receive the path from the RSI and use the path to drive around the road works.

The way the path is provided is to be determined in WP4. However, at the time of writing, the path is defined as a line with a starting point somewhere upstream of the road works, following the bus lane to the end point somewhere downstream of the road works. The RSI advices vehicles to start merging (find a gap) from the starting point onward. The distance (time) between the starting point and beginning of the road works can be updated based on the Level of Service (LOS). When vehicles reach the end point, normal traffic operations can be resumed (i.e. merge back to the rightmost non-bus lane).

Note that a ToC will still occur since AVs cannot receive the path from the RSI (since AVs by definition are lacking the ability of cooperative behaviour using communication) and must give control to human drivers.

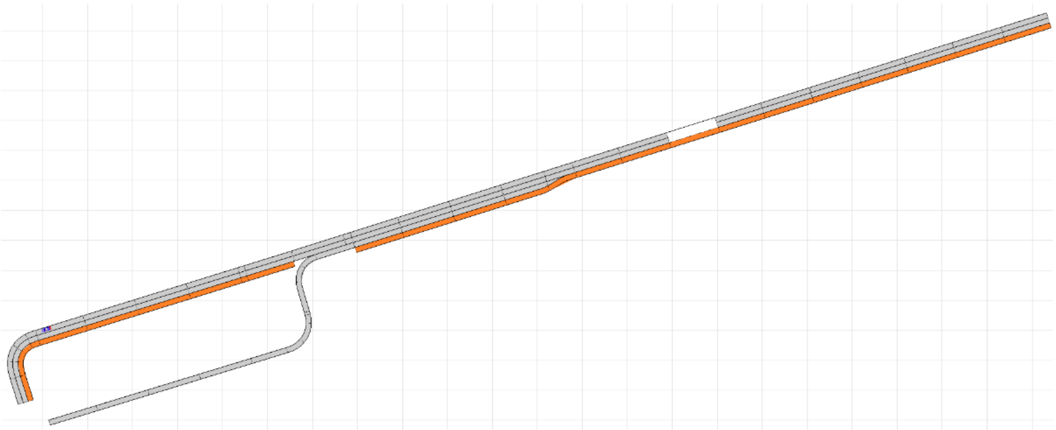
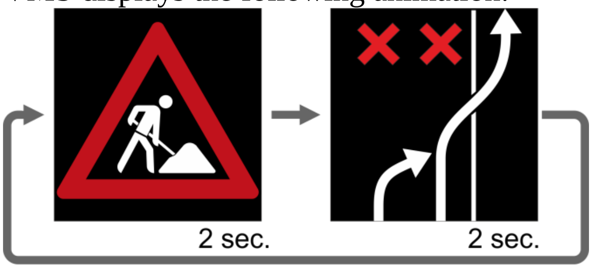
In general, all vehicles must be informed (through conventional signalling or ITS-G5) about the road works in advance to ensure there is enough time to execute lane changes and/or transitions of control without negatively affecting the traffic flow or safety.

3.2.2.2 Scenario setup

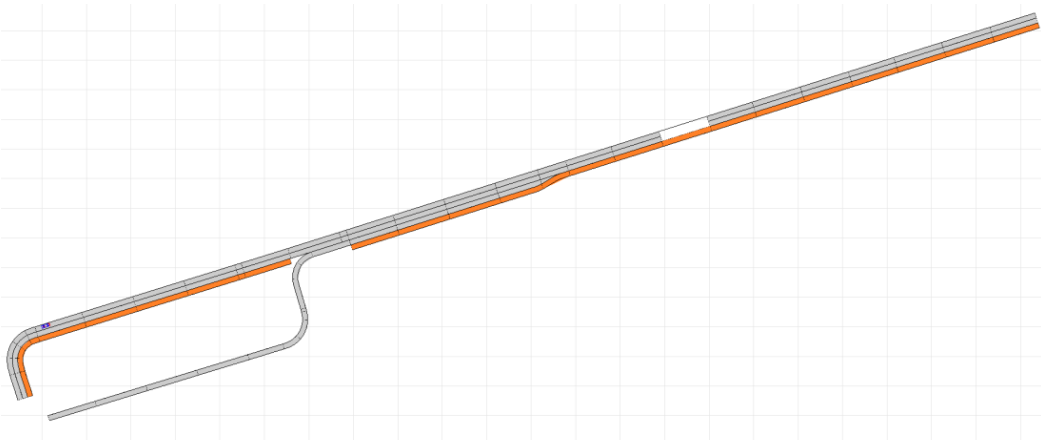
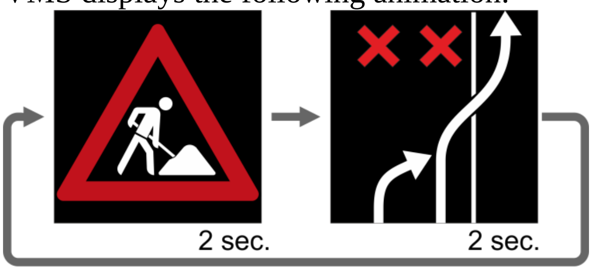
For scenario 1.1, three different tests are performed. They are summarized in the following.

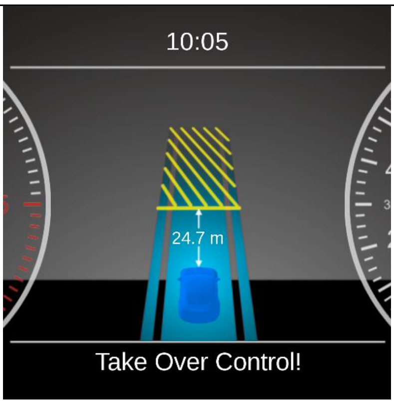
3.2.2.2.1 Test 1.1_0: “Baseline: ToC in front of blockage”

Goal	Demonstrate negative effect of a ToC in front of the blockage when no TransAID measure is applied. Successful ToC to driver. This is a V2X Day-1 test case.
Used vehicles	ViewCar2
Used	VMS, RSU

infrastructure	
Used messages	DENM, CAM
Initial situation	ViewCar2 starts on two-lane rural road, heading for a road blockage covering both lanes. Emergency/restricted lane is existing. 
Scenario script	1. VMS displays the following animation:  2. RSU sends DENM::roadWorksAlert:: • closedLanes • DrivingLanesStatus • EventPosition 3. The tactical planner draws conclusion that all lanes are blocked and ToC/MRM required. 4. AutomationLevelController triggers HMI  5. Driver successfully takes over.
Associated to Requirement(s)	• REQ_I_I1_S1.1_1 • REQ_I_I1_S1.1_3

3.2.2.2.2 Test 1.1_1: "Baseline: MRM in front of blockage"

Goal	Demonstrate negative effect of a ToC in front of the blockage when no TransAID measure is applied. ToC unsuccessful. This is a V2X Day-1 test case
Used vehicles	ViewCar2
Used infrastructure	VMS, RSU
Used messages	DENM, CAM
Initial situation	ViewCar2 starts on two-lane rural road, heading for a road blockage covering both lanes. Emergency/restricted lane is existing. 
Scenario script	1. VMS displays the following animation:  2. RSU sends DENM::roadWorksAlert:: • closedLanes • DrivingLanesStatus • EventPosition 3. The tactical planner draws conclusion that all lanes are blocked and ToC/MRM required. 4. AutomationLevelController triggers HMI:

<table>
<tr><td></td><td>


5. Since the driver is not responding, the Minimum Risk Maneuver is executed by ViewCar2, resulting in a stopped vehicle on the left lane. The following HMI is shown:

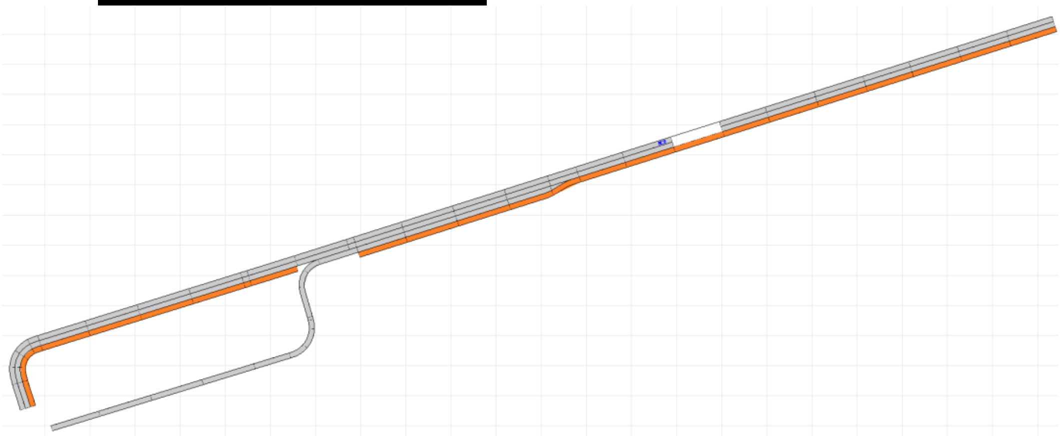
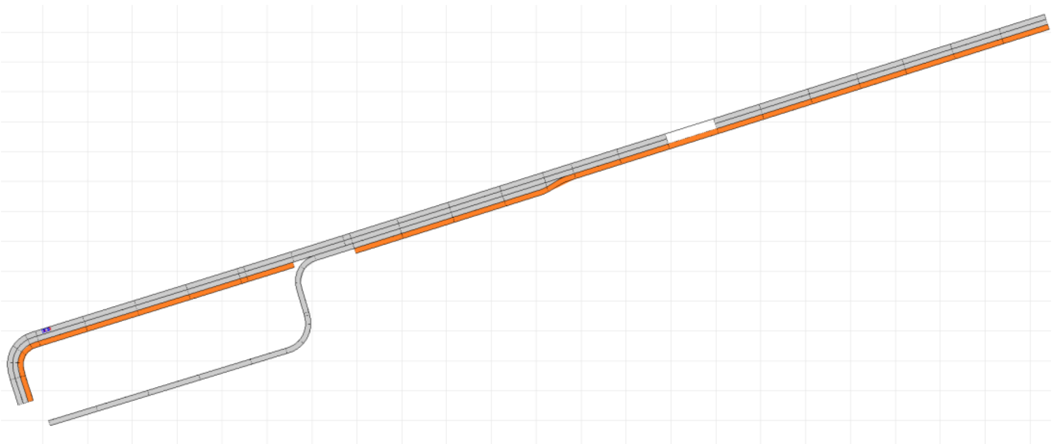
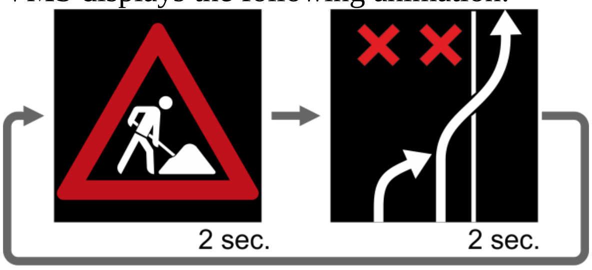
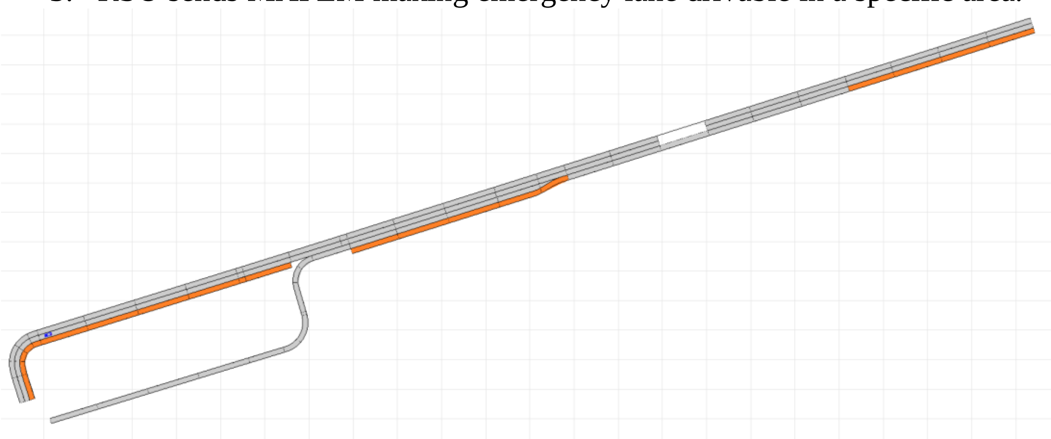
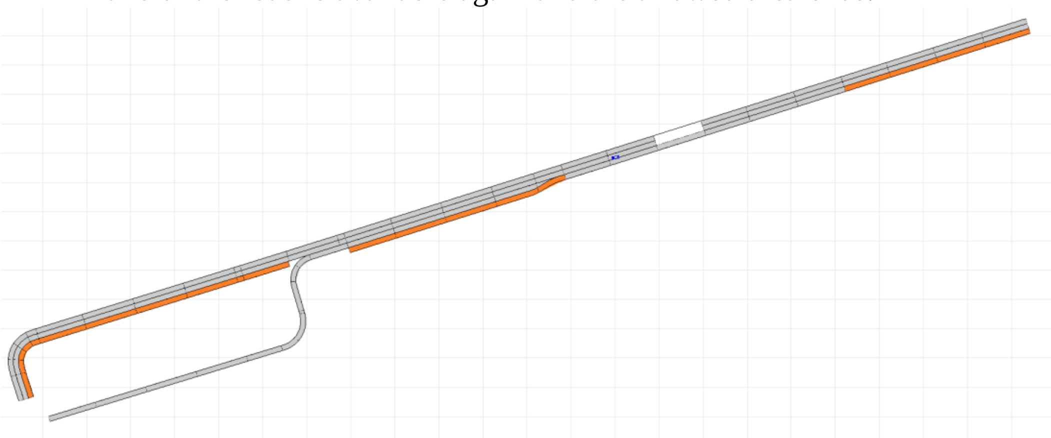

</td></tr>
<tr><td>Associated to Requirement(s)</td><td>
• REQ_I_I1_S1.1_1
• REQ_I_I1_S1.1_3
</td></tr>
</table>

Figure 16: Execution of the MRM inside the ViewCar2. Blockage indicated in digital map by DENM reception at the position of the cones on the road, left side.

3.2.2.2.3 Test 1.1_2: "Path information around blockage"

Goal	Demonstrate that infrastructure advice allows CAV to continue driving without ToC around the obstacle.
Used vehicles	ViewCar2
Used infrastructure	VMS, RSU
Used messages	DENM, CAM, MAPEM
Initial situation	ViewCar2 starts on two-lane rural road, heading for a road blockage covering both lanes. Emergency/restricted lane is existing. 

Scenario script	1. VMS displays the following animation:  2. RSU sends DENM::roadWorksAlert:: • closedLanes • DrivingLanesStatus • EventPosition 3. RSU sends MAPEM making emergency lane drivable in a specific area.  4. The digital map provider receives MAPEM, generates new borders and forwards this information to the vehicle automation 5. The tactical planner draws conclusion that lane blocks can be avoided by newly available lane. 6. ViewCar2 changes to the emergency lane and continues until the right lane of the road is available again and the allowed area ends.  7. ViewCar2 changes back to the right driving lane.

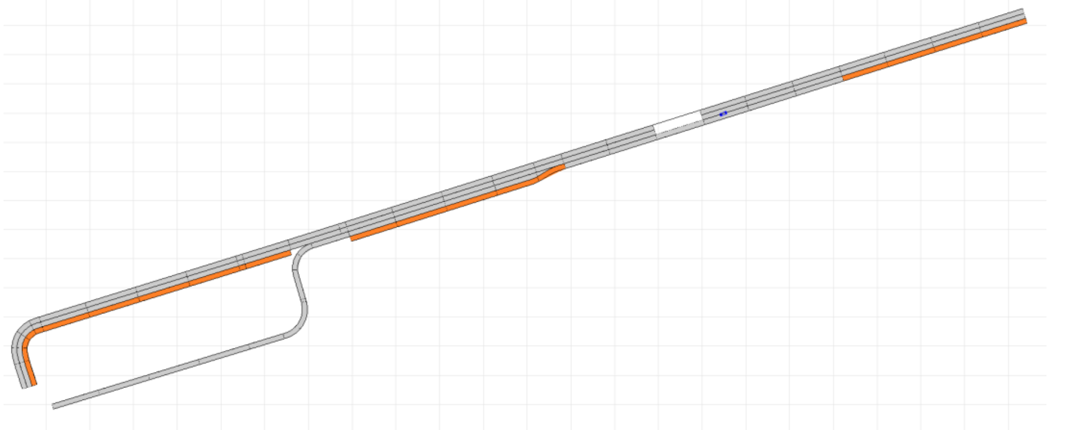
	
Associated to Requirement(s)	• REQ_V_I1_S1.1_1 • REQ_I_I1_S1.1_1 • REQ_I_I1_S1.1_2 • REQ_I_I1_S1.1_3

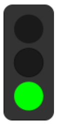
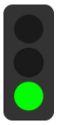
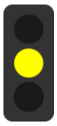
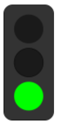
Figure 17: ViewCar2 executing the lane change around the blockage.

3.2.2.3 Feasibility results

3.2.2.3.1 Requirements verification

In addition to the feasibility assessment of the general requirements shown in section 3.2.1 a few service-specific requirements needed to be verified:

Service-specific requirement description	Req. Name	Associated Test cases successfully executed	Notes

Vehicle requirements	*Path reception* The vehicle automation shall be able to receive a path and to take it into account during trajectory planning. Of course, the final decision to follow the path is up to the automation itself. The path may be represented either as allowance to use the bus lane or as precise path containing points on the road the vehicle should pass. This will be defined later on in WP4, and WP5 is going to define the communication protocol to be used.	REQ_V_I1_S1.1_1		The path was correctly received in the format defined by D5.1. this guaranteed the successful execution all the associated test cases, hence the verification of this requirement
Infrastrucutre requirements	*Road network* The road network needs to include an explicit bus lane. This lane must be marked as non-usable in the corresponding map. In addition, road works are needed, i.e. an area which is separated on the street	REQ_I_I1_S1.1_1		Inside of the map (debug screen) there was an explicit bus lane marked in orange that represents CAVs are not allowed to use it, and road works are marked as empty road segments (white blocks). In a series production visualization/HMI, different colours/markings and/or an annotation would be used to clearly distinguish those paths.
	Sensors In order to plan valid paths it is recommended that the traffic is monitored. Positions of non-cooperative vehicles need to be included, and therefore corresponding sensors (i.e. a camera or induction loop sensors) should be used. This esp. includes the detection of stopped vehicles, either in case of Minimum Risk Manoeuvres or in case of simple traffic congestion.	REQ_I_I1_S1.1_2		RSU was equipped with a hemispherical camera, which runs an object detection algorithm to detect, classify and track objects. Transmitted CPMs were not used by the test vehicle in the first test case iteration.
	Variable Message Signs (VMS) Variable Message Signs may be used to communicate the plans of the infrastructure to the non-cooperative vehicles. Those signs should be linked to the signs signalling the road works and the lane merging. In case a (C)AV is performing a Minimum Risk Manoeuvre in this area, the sign may also be used to show warning or jam messages, see Service 4.	REQ_I_I1_S1.1_3		A VMS installed on a trailer was used during all tests displaying different signs / messages according to the tested scenario.

Requirements were followed. The reception and transmission of required V2X message was verified using a V2X module (CohdaWireless mk5) present on the test track; an external one used to sniff all V2X messages in the scenario. The capture logs show that the RSU correctly formats DENM and MAPEM messages and the content of these messages fits to the specific requirements of the tests under evaluation. The capture logs also show that the vehicle transmits frequently CAM messages, which are formatted following ETSI ITS standards. The content of the CAM is not changing dynamically though, but this was not needed for the successful execution of the tests. The test vehicle was equipped with a system status display showing the current vehicle positions on a HD map which was generated by DLR for the test track.

3.2.2.3.2 User experience

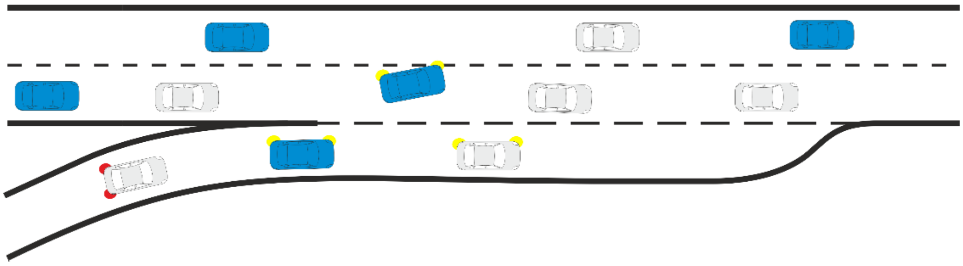
User demands were fulfilled; all test scenarios were successfully executed and serve as baseline for following use cases. This was verified by traveling as passenger in the DLR test vehicle. General user experience comments and results are covered in section 3.2.1.3.

3.2.2.3.3 Check of overall feasibility

The tested scenarios in this section build a baseline, which perform the required tasks in a reasonable and efficient way. General feasibility results from section 3.2.1.4 also apply here.

3.2.3 Requirements of Scenario 2.1: Prevent ToC/MRM by providing speed, headway and/or lane advice

3.2.3.1 Description of the scenario from D2.2



CAVs, AVs, CVs, and LVs drive along a motorway merge segment or enter the mainline motorway lanes through an on-ramp. The RSI monitors traffic operations along the motorway merge segment and detects the available gaps on the right-most mainline lane to estimate speed and lane advice for merging CAVs and CVs coming from the on-ramp. The scenario assumes that CAVs and CVs continuously update their speed and lane information to the RSI (in a near-real-time fashion). In addition, the RSI also fuses this information with measurements obtained via available road-side sensors. The speeds and locations of AVs and LVs can be estimated based on the information gathered via the latter sensors and the location (and available sensing information) of the other vehicles (being CAVs or CVs). This scenario necessitates the exchange of the required types of messages (i.e. CPM/CAM/MCM).

The central core of this scenario is the guidance towards or creation of gaps in the motorway's right-most lane (that is not part of the on-ramp). If the available gaps there are not large enough to allow the safe and smooth merging of on-ramp vehicles, speed and lane advice are also provided to the CAVs and CVs driving there, thereby creating the necessary gaps in traffic to facilitate the smooth merging of on-ramp vehicles. Thus, gaps are created by the exchange of suitable lane change advices to these two kinds of vehicles; AVs and LVs do not receive information. Note that we do not adopt explicit ramp-metering algorithms to control the average in-flow of vehicles to the motorway. The ramp meter will only be used to assist vehicles in entering the motorway at the right moment, but not to restrict in-flow more than in the baseline. In addition, advice to vehicles is only given within a certain action-zone, i.e. upstream of and at the merge location. Beyond that, further downstream, vehicles can default back to their previous own behaviour.

Without the aforementioned measures vehicles might be impeded or involved in safety critical situations under specific traffic conditions (e.g. incidents) or automated driving operations (e.g. platooning at motorway merge/diverge segments). Under these circumstances, automated vehicles might request ToCs or execute MRMs for safety reasons.

Note: aggressive lane changes of human drivers can disturb traffic flow and cause emergency breaks or high decelerations. These do not pose great risks in free-flowing traffic, as the traffic streams remain locally and asymptotically stable (initial finite disturbances exponentially die out, even along CAV platoons). However, the more congested traffic becomes, the higher the instability of a traffic stream gets. Hence, such local disturbances are not smoothed out anymore, resulting in sudden and drastic changes in the speed profiles of upstream vehicles. Similarly, lane changes of slow vehicles (e.g. trucks) have a higher impact, since they require larger gaps and can force other vehicles to suddenly break. Compared to cars, truck lane changes are minor in occurrence (if not forbidden by traffic law). However, in case they do occur, they typically lead to 'moving bottlenecks' due to their lower average speeds, especially in free-flow and synchronised traffic flows. Another situation, in which truck lane changes are more frequent, is when a truck enters the *motorway via an on-ramp and trucks on the main motorway provide spacing by moving out of the way, creating again the aforementioned moving bottleneck.*

3.2.3.2 Scenario setup

For scenario 2.1, six different tests are performed. They are summarized in the following. It has to be remarked that the original scenario 2.1 does only include speed advice to the vehicle on the ramp (Test 2.1_5). Nevertheless, it was defined that advice could basically also be given to the vehicles on the highway, either for speed or for preferred lane usage. These aspects will be further investigated during the second project iteration, and also be covered in the simulation activities later on in the other work packages.

3.2.3.2.1 Test 2.1_0: "Baseline: Ramp without communication"

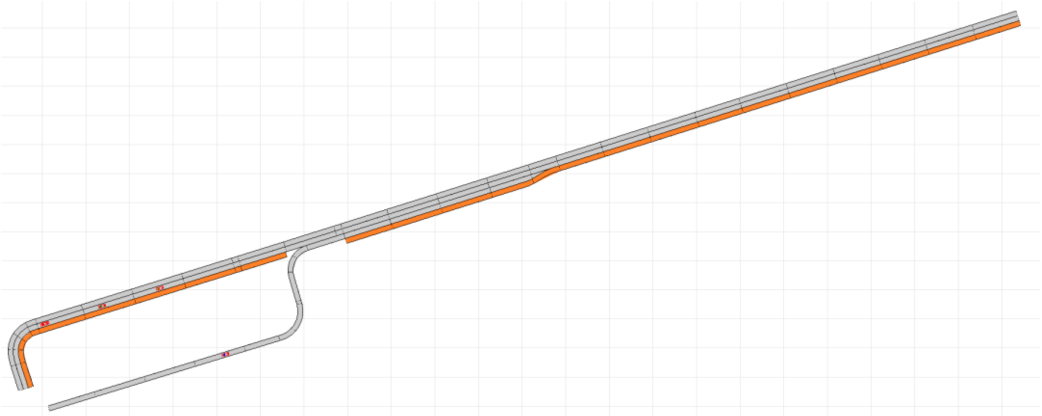
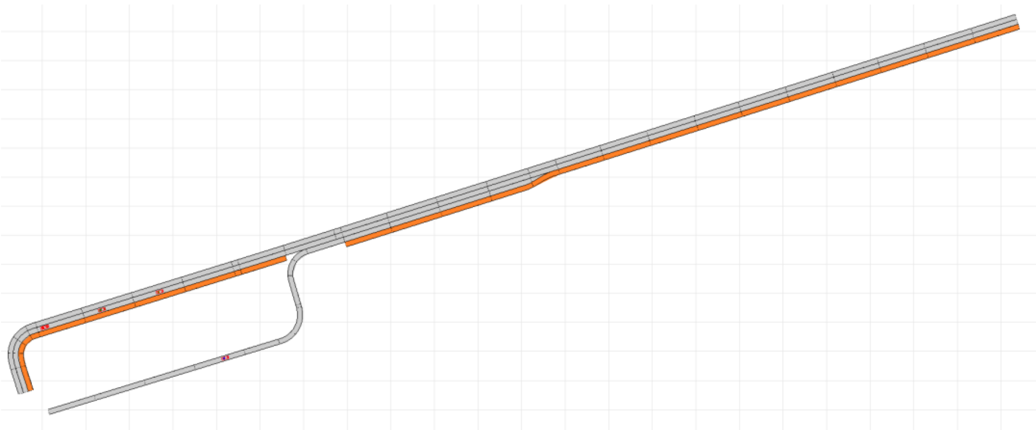
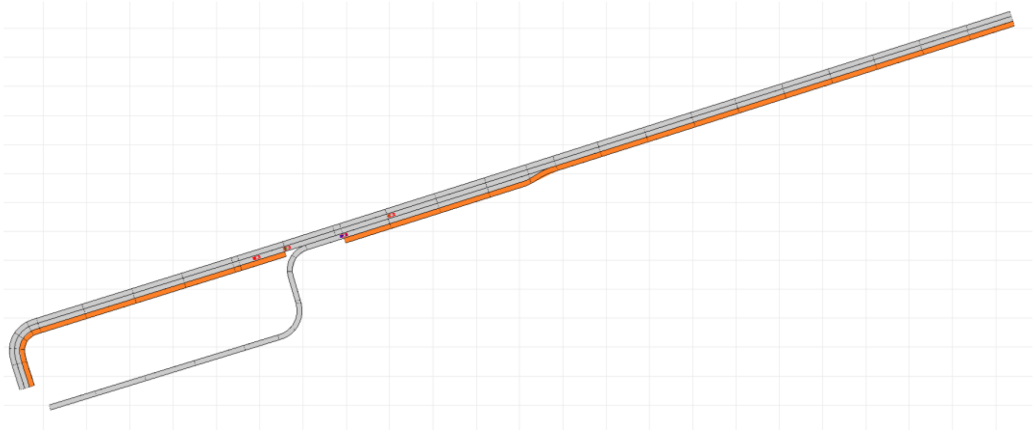
Goal	Demonstrate negative effect of a CAV not able to merge from a ramp to a highway.
Used vehicles	ViewCar2, FASCarE, T5 bus, optional legacy vehicle
Used infrastructure	None
Used messages	CAM
Initial situation	ViewCar2 starts on ramp entering highway. Several vehicles drive on the right lane of the highway close to each other 
Scenario script	1. When trying to enter the highway, no gap is found. Vehicle is braking and waiting until sufficient gap available
Associated to Requirement(s)	Only generic requirements

Figure 18: Blocked entrance, as manually driven FASCarE on adjacent lane does not allow merging (driving at the same speed). Lane change not possible, ViewCar2 stops.

3.2.3.2.2 Test 2.1_1: "Cooperative lane change: Vehicle on highway opens gap"

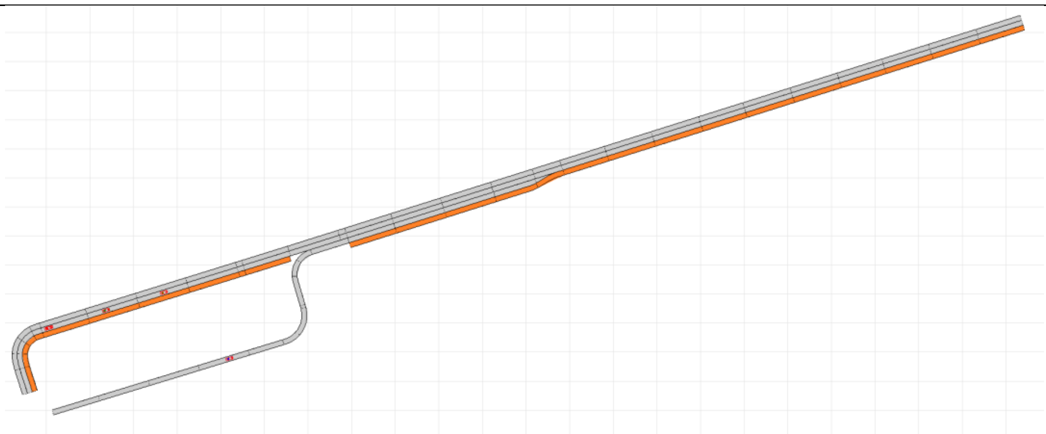
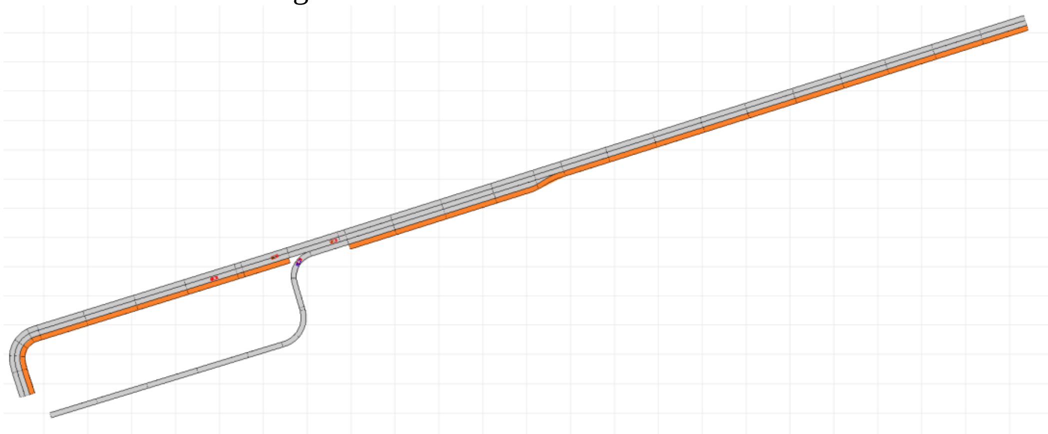
Goal	Demonstrate abilities of cooperative lane change without infrastructure support. Here, the vehicle on the highway opens a gap by braking.
Used vehicles	ViewCar2, FASCarE, T5 bus, optional legacy vehicle
Used infrastructure	None
Used messages	CAM, V2V-MCM
Initial situation	ViewCar2 starts on ramp entering highway. Several vehicles, including FASCarE as other CAV drive on the right lane of the highway close to each other 

<table>
<tr><td>Scenario script</td><td>1. When ViewCar2 arrives at the entrance, it indicates its desire to change to the right lane of the road via MCM::VMC::DesiredTrajectory
2. FASCarE on highway indicates cooperation by sending MCM::VMC::PlannedTrajectory containing a braking trajectory
3. FASCarE brakes

4. ViewCar2 enters highway in new gap</td></tr>
<tr><td>Associated to Requirement(s)</td><td>Only generic requirements.</td></tr>
</table>

Note: This test case has only been executed in simulation during the first iteration, since the message implementation at DLR was delayed and testing was impossible before deliverable submission. The tests will be repeated and the results included in the second iteration version of this deliverable.

3.2.3.2.3 Test 2.1_2: "Cooperative lane change: Vehicle on highway changes lane"

Goal	Demonstrate abilities of cooperative lane change without infrastructure support. Here, the vehicle on the highway opens a gap by changing lane.
Used vehicles	ViewCar2, FASCarE, T5 bus, optional legacy vehicle
Used infrastructure	None
Used messages	CAM, V2V-MCM
Initial situation	ViewCar2 starts on ramp entering highway. Several vehicles, including FASCarE as other CAV drive on the right lane of the highway close to each other

<table>
<tr><td></td><td></td></tr>
<tr><td>Scenario script</td><td>1. When ViewCar2 arrives at the entrance, it indicates its desire to change to the right lane of the road via MCM::VMC::DesiredTrajectory
2. FASCarE on highway indicates cooperation by sending MCM::VMC::PlannedTrajectory containing a lane change to the left
3. FASCarE changes lane

4. ViewCar2 enters highway in new gap</td></tr>
<tr><td>Associated to Requirement(s)</td><td>Only generic requirements</td></tr>
</table>

Note: This test case has only been executed in simulation during the first iteration, since the message implementation at DLR was delayed and testing was impossible before deliverable submission. The tests will be repeated and the results included in the second iteration version of this deliverable.

3.2.3.2.4 Test 2.1_3: “Ramp assist: Infrastructure advices vehicle on highway to change lane”

Goal	Demonstrate abilities of infrastructure support. Here, the infrastructure advises individual vehicles on the highway to change lane.
Used vehicles	ViewCar2, FASCarE, T5 bus, optional legacy vehicle
Used infrastructure	RSU, Camera
Used messages	CAM, I2V-MCM

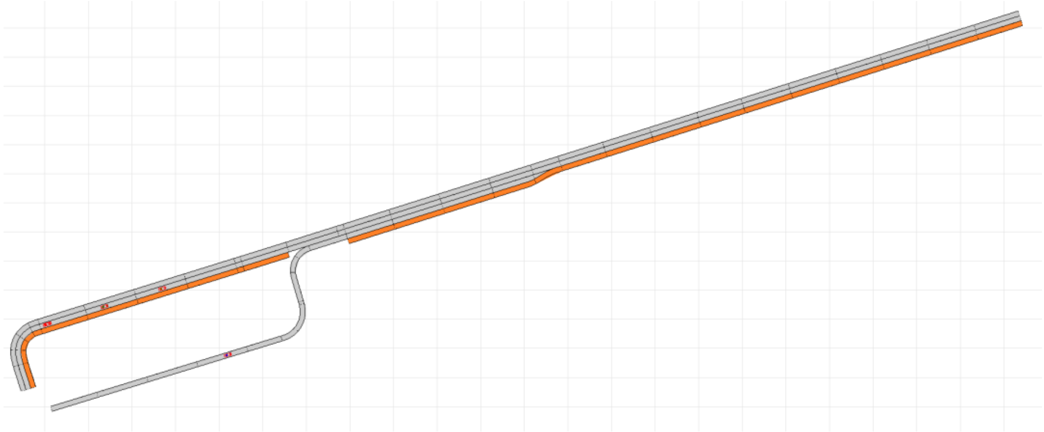
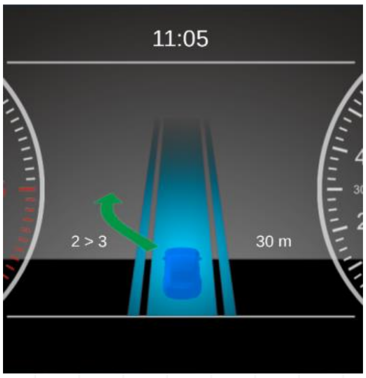
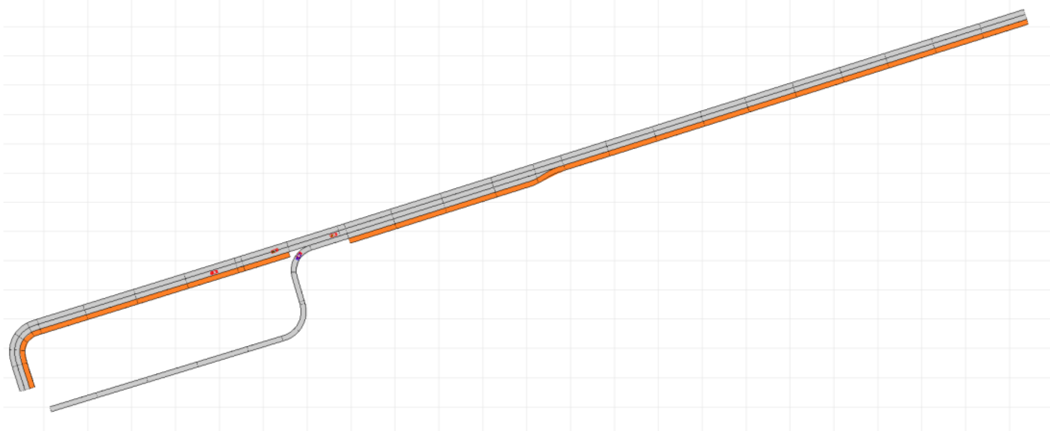
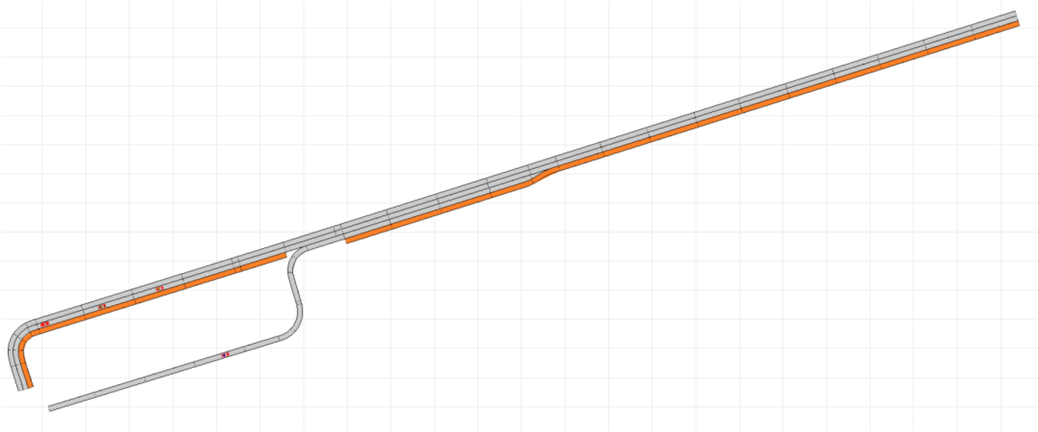
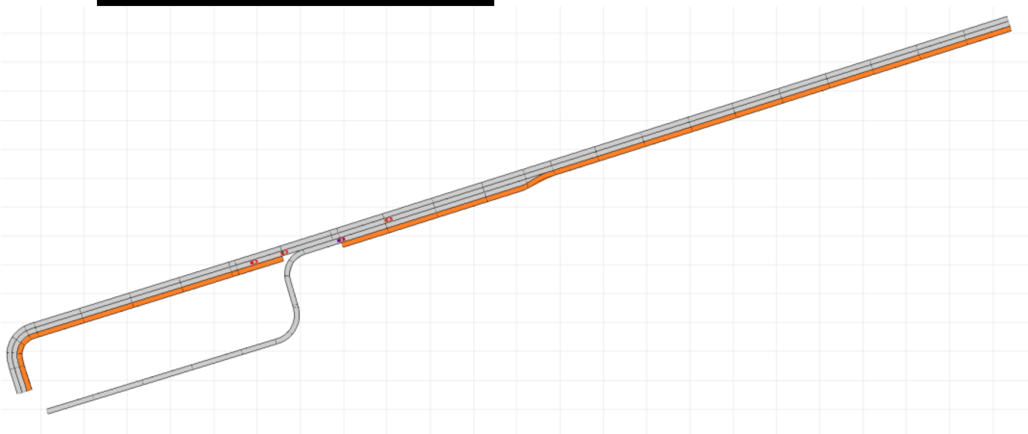
Initial situation	ViewCar2 starts on ramp entering highway. Several vehicles, including FASCarE as other CAV drive on the right lane of the highway close to each other 
Scenario script	1. RSU sends MCM:: RSU_SMC ::LaneAdvice=3 with distance=150 to FASCarE driving on right lane (=2) of the highway 2. FASCarE follows advice and changes lane. The following HMI is shown:   3. When ViewCar2 arrives at the entrance, it is able to perform the lane change into the new gap
Associated to Requirement(s)	• REQ_V_I1_S2.1_2 • REQ_I_I1_S2.1_1 • REQ_I_I1_S2.1_2

Figure 19: Image taken of the test while FASCarE automatically performs lane change.

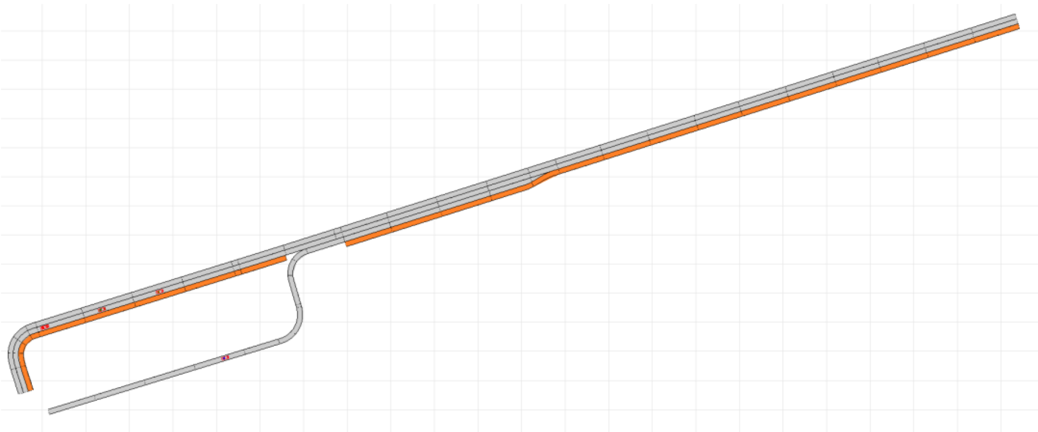
Figure 20: After successful lane change of the FASCarE, the ViewCar2 merges onto the highway.

3.2.3.2.5 Test 2.1_4: "Ramp assist: Infrastructure advices vehicle on highway to change speed"

Goal	Demonstrate abilities of infrastructure support. Here, the infrastructure advices individual vehicles on the highway to change speed.
Used vehicles	ViewCar2, FASCarE, T5 bus, optional legacy vehicle
Used infrastructure	RSU, Camera
Used messages	CAM, I2V-MCM

Initial situation	ViewCar2 starts on ramp entering highway. Several vehicles, including FASCarE as other CAV drive on the right lane of the highway close to each other 
Scenario script	1. RSU sends MCM::RSU_SMC::CarFollowingAdvice::DesiredBehavior:: TargetSpeed=9 to FASCarE driving on right lane of the highway 2. FASCarE follows advice and slows down to open gap. The following HMI is shown:   3. When ViewCar2 arrives at the entrance, it is able to perform the lane change into the new gap in front of FASCarE.
Associated to Requirement(s)	• REQ_V_I1_S2.1_1 • REQ_I_I1_S2.1_1 • REQ_I_I1_S2.1_2

3.2.3.2.6 Test 2.1_5: "Ramp assist: Infrastructure advices vehicle on ramp to change speed"

Goal	Demonstrate abilities of infrastructure support. Here, the infrastructure advices individual vehicles on the highway to change speed.
Used vehicles	ViewCar2, FASCarE, T5 bus, optional legacy vehicle
Used infrastructure	RSU, Camera
Used messages	CAM, I2V-MCM
Initial situation	ViewCar2 starts on ramp entering highway. Several vehicles, including FASCarE as other CAV drive on the right lane of the highway close to each other 
Scenario script	1. RSU sends MCM::RSU_SMC::CarFollowingAdvice::DesiredBehavior:: TargetSpeed=9 to ViewCar2 driving on ramp 2. ViewCar2 follows advice and slows down. The following HMI is shown:  3. When ViewCar2 arrives at the entrance, it is able to perform the lane change behind FASCarE.

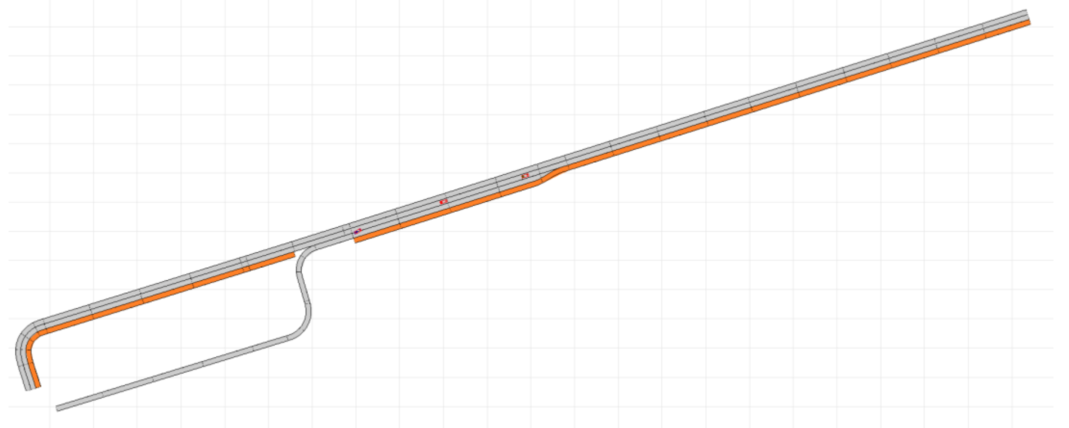
	
Associated to Requirement(s)	• REQ_V_I1_S2.1_1 • REQ_I_I1_S2.1_1 • REQ_I_I1_S2.1_2

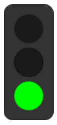
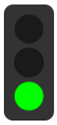
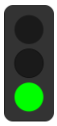
Figure 21: Successful lane change after speed adaptation of ViewCar2 on the ramp.

3.2.3.3 Feasibility results

3.2.3.3.1 Requirements verification

In addition to the feasibility assessment of the general requirements shown in section 3.2.1 a few service-specific requirements needed to be verified:

Service-specific requirement description	Req. Name	Associated Test cases successfully executed	Notes

Vehicle requirements	*Speed advice following* The CAVs/CVs need to be able to receive speed advice from the infrastructure. In case of a CAV, the advice needs to be taken into account during trajectory planning, although the vehicle automation itself has the right to overrule the advice. In case of a CV, the speed advice is forwarded to the driver with an appropriate HMI.	REQ_V_I1_S2.1_1		Speed advice received and followed by test vehicle.
	Lane advice following Also, lane advice needs to be received and taken into account in the same way then speed advice.	REQ_V_I1_S2.1_2		Lane advice received and followed. HMI shows target lane using moving arrows inside of the cluster display
Infrastructure requirements	*Speed and lane advice generation* The infrastructure must be able to generate speed and lane advice based on the detected situation and disseminate them using an RSU.	REQ_I_I1_S2.1_1		RSU generated advice that was received by test vehicle as well as other V2X receivers present on the test area. Howewver, the advice was not generated based on the situation detected by the RSU.
	Sensors This scenario requires very precise detection of vehicles and vehicle behaviour, as probable gaps have to be estimated early enough to provide appropriate advice to the vehicles.	REQ_I_I1_S2.1_2		RSU with dedicated camera detected surrounding objects (road users) and transmitted these using CPMs

For this set of tests, the reception and transmission of required V2X messages was also verified using the V2X module (CohdaWireless mk5) present on the test track. The capture logs show that the RSU correctly formats MCM messages following the TransAID MCM ASN.1 definition, and the content of these messages fits to the specific requirements of the tests under evaluation. In particular, the captured messages show the RSU´s lane change and car following advice that are addressed to the vehicle on the highway and/or ramp depending on the test.

3.2.3.3.2 User experience

As already mentioned in section 3.2.1.3 vehicle speeds and acceleration/deceleration are fine. Also here a clear HMI supports travel comfort and perceived safety for passengers. Especially before and during lane changes (from ramp to highway) it must be easily recognizable that surrounding traffic is detected by the system (not leading to false impressions and counteractions by passengers/driver).

3.2.3.3.3 Check of overall feasibility

It can be clearly seen that advice applied to vehicles on the ramp is less disturbing the overall traffic flow compared to advice that affects vehicles traveling on the highway. For this reason, a higher priority should be given to advice at the on-ramp (which can be followed in less dense traffic). Lane changes of vehicles can have a higher impact on the overall traffic flow, which requires a constant tracking of surrounding vehicles (especially non-cooperative LV). This might lead to the strict requirement of the presence of infrastructure sensing units supporting coordinated lane change advice or an exclusion of coordinated multiple vehicle lane changes in complex road architectures (e.g. sharp turns or multiple junctions/ramps in short distances).

3.2.4 Requirements of Scenario 3.1: Apply traffic separation before motorway merging/diverging

3.2.4.1 Description of scenario from D2.2

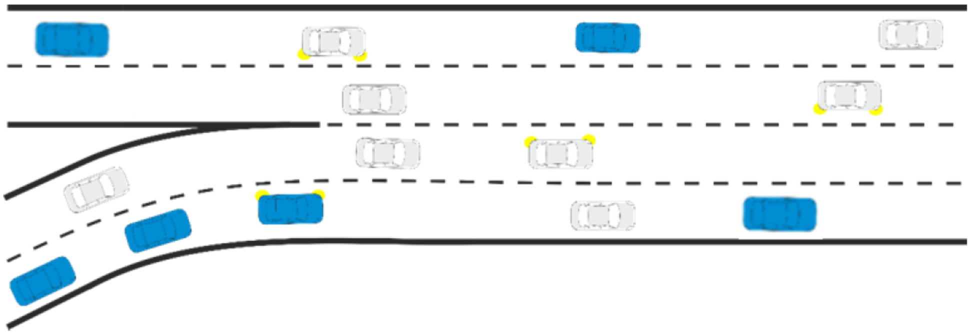


Figure 22: schematic overview of Scenario 3.1

CAVs, CAV platoons, CVs and LVs drive along two 2-lane motorways that merge into one 4-lane motorway. After the merging point, vehicles will drive to their target lane. RSI monitors the amount of different types of vehicles upstream through collective perception but also via CAM receptions, and infra sensors.

Based on the provided traffic separation policy, CAVs and CAV platoons move to the left lane of the left 2-lane motorway and to the right on the right 2-lane motorway at some point upstream of the merging point (where merging usually starts). CVs move to the other lanes not allocated to CAVs and CAV platoons. CAVs and CAV platoons thus enter the 4-lane section on the outer lanes, giving space to manually driven vehicles (CVs and LVs) to occupy the central lanes (where human driving still may generate risky situations).

Following this approach, the overall number of risky situations will be reduced which will positively affect the number of ToCs in this area.

At some point downstream of the merging point, the traffic separation is disabled, and all vehicles can gradually start changing lanes to reach their target destination.

3.2.4.2 Scenario setup

The effects of this scenario can best be seen in traffic simulation. Nevertheless, the feasibility should be shown as well. Therefore, the scenario is simplified, so that it focusses on traffic separation only. At this moment, it is not decided whether a full separation is targeted, meaning that also non-cooperative vehicles should change to their dedicated lane, or if the separation is only involving cooperative vehicles, separating CAVs and AVs to one lane and CVs and LVs to the other. A decision will be made after the baseline simulations have been performed.

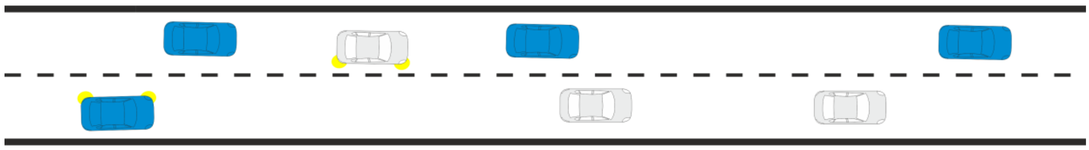


Figure 23: Schematic overview of the simplified Scenario 3.1

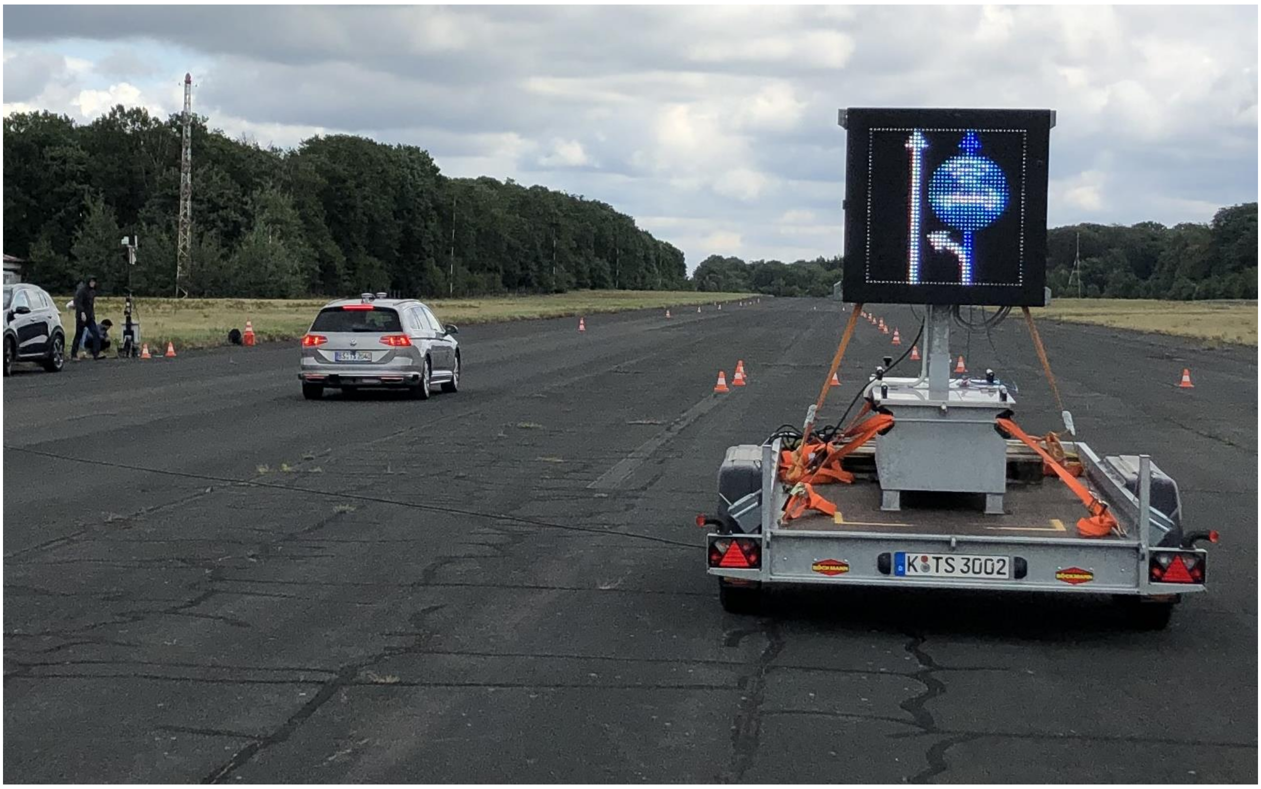
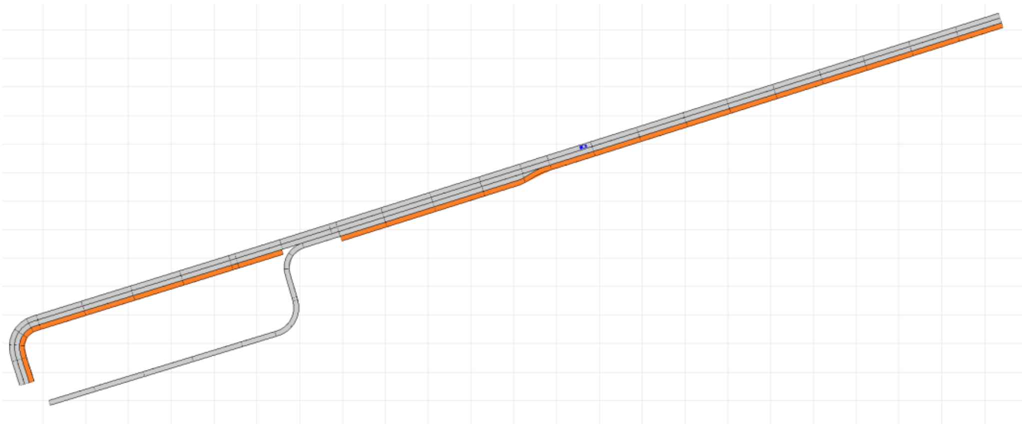
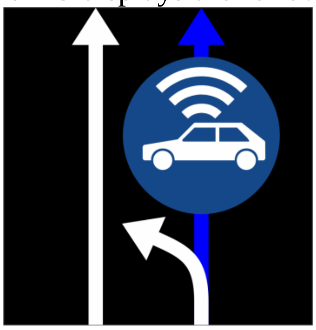


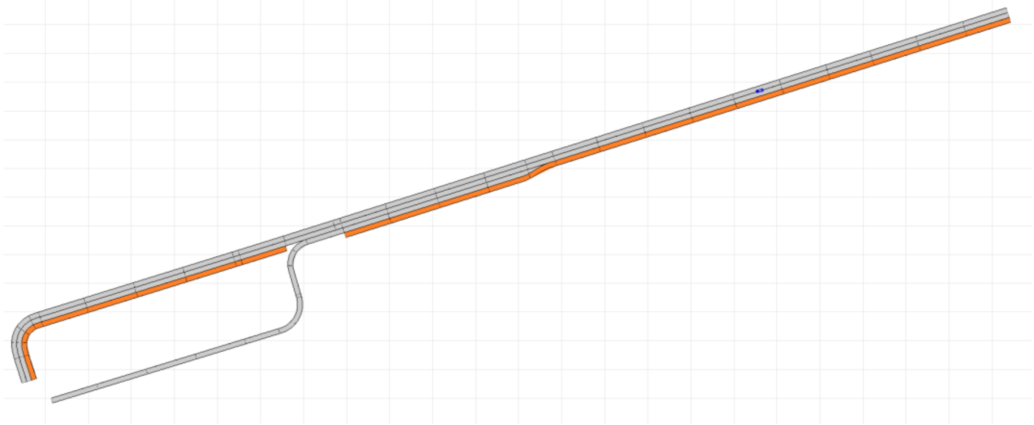
Figure 24: DLR's ViewCar2 executing the scenario 3.1 tests

Figure 25: Internal view showing the received lane advice to the right lane in the cluster instrument

For the simplified scenario 3.1, one single test is performed, described in the following.

3.2.4.2.1 Test 3.1_0: "Traffic separation by lane advices"

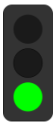
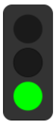
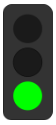
Goal	Demonstrate the ability to perform traffic separation by receiving appropriate messages.
Used vehicles	ViewCar2
Used infrastructure	RSU, Camera, VMS
Used messages	CAM, I2V-MCM
Initial situation	ViewCar2 starts on left lane of highway. Other legacy vehicles optionally drive on the right lane of the highway 
Scenario script	1. VMS displays the following sign:  2. RSU sends MCM:: RSU_SMC ::LaneAdvice=2 with distance=500m to ViewCar2 3. ViewCar2 changes to right lane

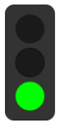
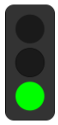
	 4. Legacy vehicle changes to left lane
Associated to Requirement(s)	• REQ_V_I1_S3.1_1 • REQ_V_I1_S3.1_2 • REQ_I_I1_S3.1_1 • REQ_I_I1_S3.1_2 • REQ_I_I1_S3.1_3

3.2.4.3 Feasibility results

3.2.4.3.1 Requirements verification

In addition to the feasibility assessment of the general requirements shown in section 3.2.1 a few service-specific requirements needed to be verified:

Service-specific requirement description		Req. Name	Associated Test cases successfully executed	Notes
Vehicle requirements	*Separation advice following* The CAVs/CVs need to be able to receive separation advice from the infrastructure. In case of a CAV, the advices need to be taken into account during trajectory planning, although the vehicle automation itself has the right to overrule the advice. This means that defined lanes should be marked as non-preferable. In case of a CV, the separation advice is forwarded to the driver with an appropriate HMI.	REQ_V_I1_S3.1_1		Separation advice received and followed by test vehicle. HMI not showing the reason for trajectory changes.
	Lane advice following Also, lane advice need to be received and taken into account.	REQ_V_I1_S3.1_2		Test vehicle received lane change advice and followed them accordingly by changing lane if required.
Infrastructure requirements	*Separation advice generation* The infrastructure needs to be able to generate separation advice. The advice may be simply switched on for areas on the road. No further detection capabilities are needed for the feasibility assessment, although the LOS needs to be determined to estimate whether separation needs to be done or not.	REQ_I_I1_S3.1_1		The infrastructure-generated advice to request vehicles in a specific area to separate based on their automation level. Variable Message Sign (VMS) trailer was also used to generate separation advices for LVs.

	Lane advice generation The generation of lane advice is already covered in scenario 2.1, but may also be useful in the context of scenario 3.1, which has to be defined after the baseline simulations. Please note that separation itself is not needing lane advice capabilities, but those capabilities may be an adequate additional option for the implementation of separation.	REQ_I_I1_S3.1_2		RSU provided lane advices to present vehicles. For later test it might be interesting to execute tests with multiple CAV/CVs as well as LVs.
	Variable message signs (VMS) In case non-cooperative vehicles need to be advised, variable message signs can be used to indicate the separation, e.g. by offering lane usage advices.	REQ_I_I1_S3.1_3		VMS showed a traffic separation advice sign to separate LVs and CAV/CVs.

For this scenario, the reception and transmission of required V2X message was also verified using the V2X module (CohdaWireless mk5) present on the test track. The capture log shows that the RSU correctly formats MCM messages following the TransAID MCM ASN.1 definition, and the content of these messages fits to the specific requirements of the scenario under evaluation. In particular, the captured messages show the RSU´s lane change advice including the target station that should follow the advice, where the lane change should be performed, and what is the target lane.

3.2.4.3.2 User experience

General user experience results from section 3.2.1.3 also apply here. The usage of a VMS (as here done installed on a trailer) is a reasonable, easy understandable and cost effective solution to inform drivers of LV to change the lane (e.g. no need to install a V2X reception unit in LV). Slight adaptions of the lateral speed during lane change / merge could improve the safety impression of passengers while changing from the ramp to the straight road path (impression of a slight vehicle over swing).

3.2.4.3.3 Check of overall feasibility

Due to the lack of test vehicles, the scenario couldn’t be tested to the fullest extent in the first iteration. Excluding the need to change the traffic regulations, the scenario results will be implemented with a higher spread of CV/CAVs. For this reason it seems right now not to be a feasible solution (drivers of LV could also force a break up by intentionally using wrong lanes), but it can be forecasted to be a feasible solution in future.

3.2.5 Requirements of Scenario 4.2: Safe spot in lane of blockage

3.2.5.1 Description of scenario from D2.2

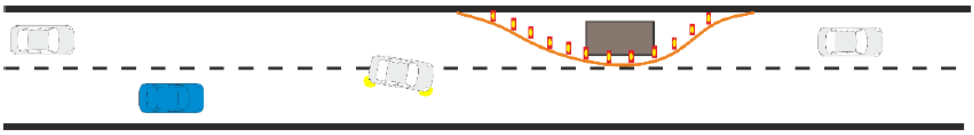


Figure 26: schematic overview of Scenario 4.2

There is a construction site covering one lane of the motorway road. The deployed RSI has information about the construction area and the vicinity of it and provides this information to the approaching CAVs.

Some CAVs are not able to pass the construction site without any additional guidance. Therefore, they need to perform a ToC. A ToC might be unsuccessful, so the respective CAV must perform an MRM. Without additional measures, the CAV would simply brake and stop on the lane it is driving, most likely disrupting the traffic flow when happening on the right lane (see figure),

To avoid this, the RSI also monitors the area just in front of the construction site and offers this place as a safe stop to the vehicle, if free. The CAV uses the safe spot information just in front of the construction site to come to a safe stop in case of an MRM.

Note: Service 4 basically is an additional measure to the other services, used when any ToC is about to fail (see D2.1 [23] *for details) and the impact of MRMs should be reduced. In this specific case of Scenario 4.2, it can be seen as an extension to Scenario 1.1.*

3.2.5.2 Scenario setup

This scenario will not be changed for the feasibility assessment. Nevertheless, discussions are going on focussing on the exact shape of safe spots. As a first idea, which is followed during the first iteration of the project, safe spots look as shown in Figure 27. Safe spots are separated areas on the road offering room for (C)AVs to stop and limited space to accelerate again. The number of the safe spots and the related size of the area are linked to the number of occurring Minimum Risk Manoeuvres, and needs to be estimated during the base line simulations. Nevertheless, it has been agreed that all safe spot related measures should include scalability, so that the derived measures apply for single safe spots as well as for larger areas.

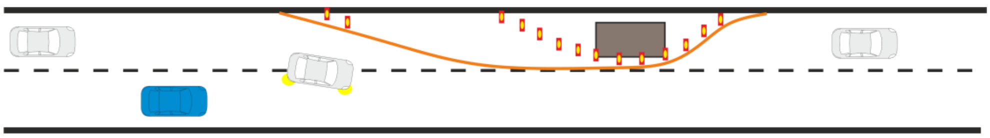


Figure 27: Safe spot design

Following the most recent discussions in WP5, an explicit reservation of safe spots is not envisioned. The infrastructure is only providing information about the free areas, and the vehicles may implicitly block the areas by sharing Manoeuvre Coordination Messages. The final decision is described in D5.1. In case two or more vehicles decide to make use of the same safe spot at the very same time, the conflict will be visible right after sharing the Manoeuvre Coordination Message. If one of the vehicles is not able to use another safe spot, or if there is no other available, the vehicle is going to stop on the road as it would do without the TransAID measure.

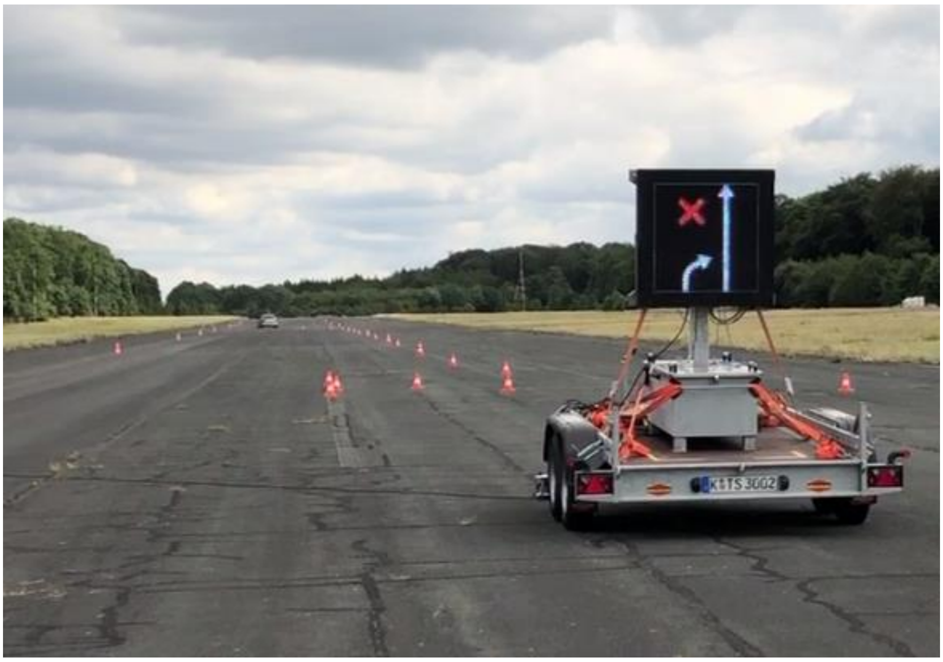
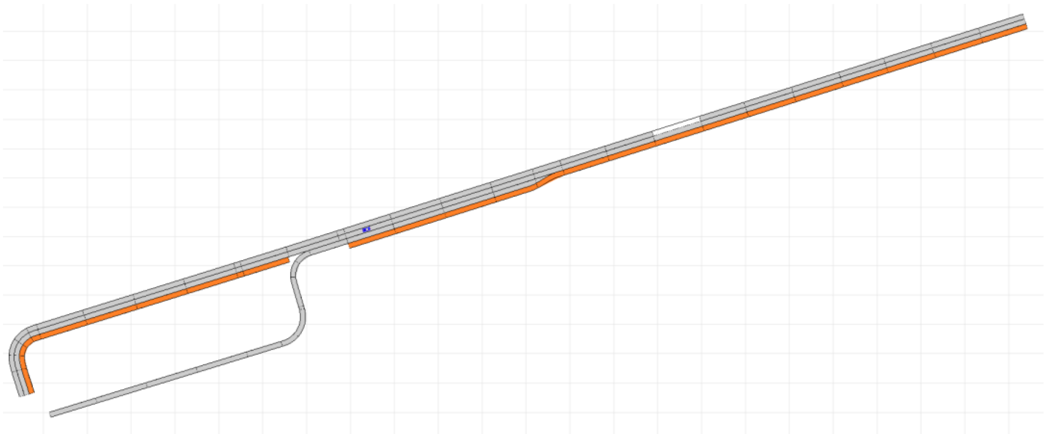


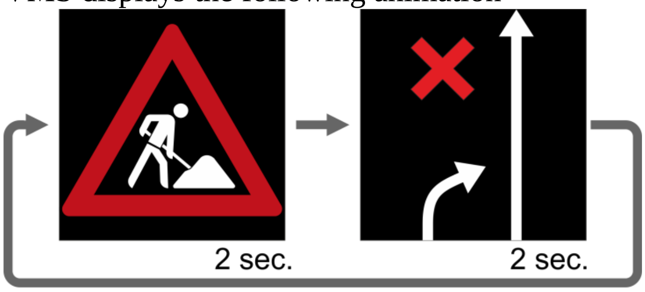
Figure 28: DLR's ViewCar2 executing the scenario 4.2 tests.

For scenario 4.2, three tests are performed, described in the following.

3.2.5.2.1 Test 4.2_0: "Baseline: MRM on free lane"

Goal	Demonstrate negative effects of a ToC on the right lane in front of the blockage when no TransAID measure is applied. ToC unsuccessful, MRM executed.
Used vehicles	ViewCar2
Used infrastructure	RSU, VMS
Used messages	CAM, DENM
Initial situation	ViewCar2 starts on right lane of two-lane road, followed by a legacy vehicle. The left lane of the road is blocked by roadworks. The vehicle is by default not able to pass the roadworks on the right lane. 

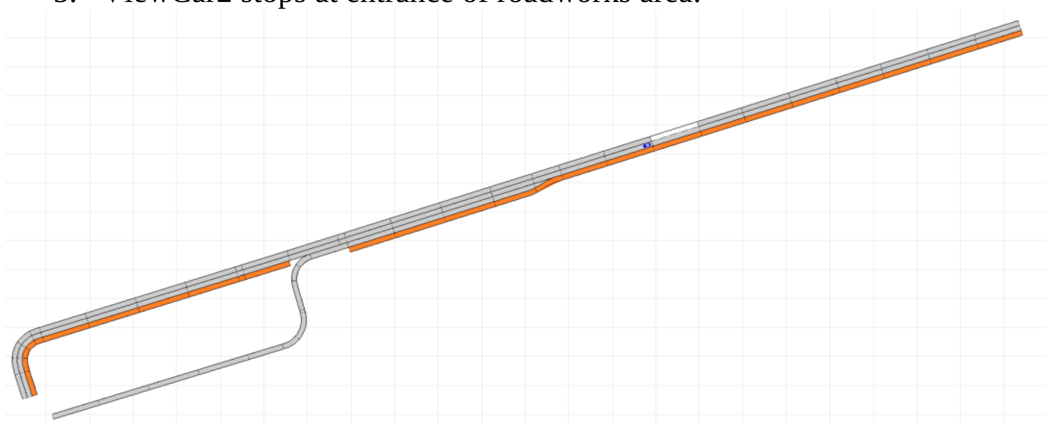
<table>
<tr>
<td>Scenario script</td>
<td>
1. VMS displays the following animation



2. RSU broadcasts DENM::Roadworks
3. ViewCar2 issues ToC. The following HMI is shown:

4. Driver does not take over control, standard MRM is executed. The following HMI is shown:

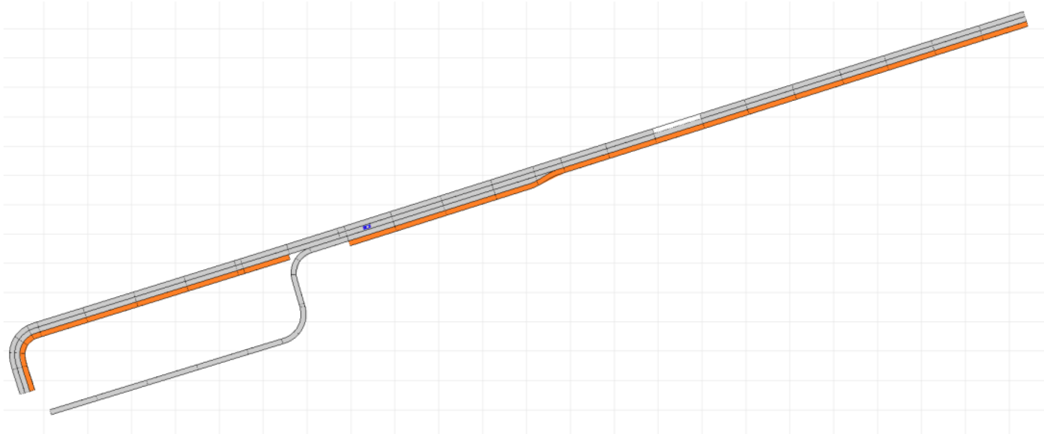
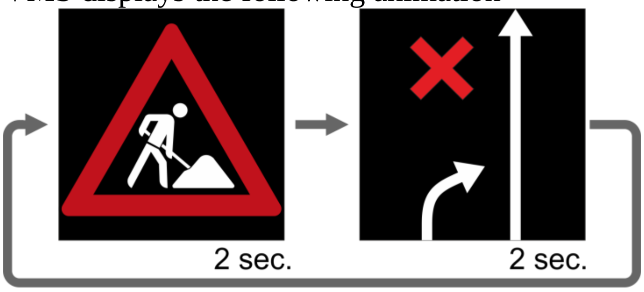
5. ViewCar2 stops at entrance of roadworks area.


</td>
</tr>
<tr>
<td>Associated to Requirement(s)</td>
<td>

- REQ_V_I1_S4.2_1
- REQ_V_I1_S4.2_2
- REQ_I_I1_S4.2_1
</td>
</tr>
</table>

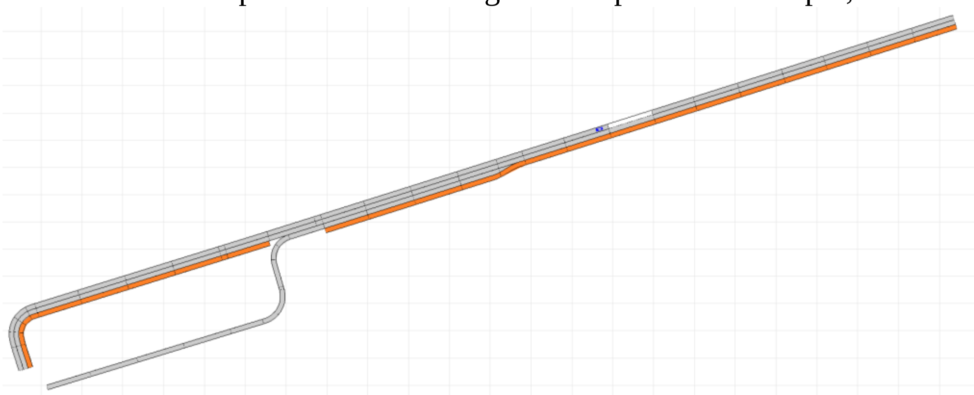
	• REQ_I_I1_S4.2_2

3.2.5.2.2 Test 4.2_1: "MRM into SafeSpot on Left Lane"

Goal	Demonstrate benefits of performing a Minimum Risk Maneuver into a Safe Spot in front of the roadworks area.
Used vehicles	ViewCar2
Used infrastructure	RSU, Camera, VMS
Used messages	CAM, DENM, I2V-MCM
Initial situation	ViewCar2 starts on right lane of two-lane road, followed by a legacy vehicle. The left lane of the road is blocked by roadworks. The vehicle is by default not able to pass the roadworks on the right lane. 
Scenario script	1. VMS displays the following animation  2. RSU broadcasts DENM::Roadworks 3. RSU sends MCM:: RSU_SMC ::ToCAdvice to ViewCar2 4. ViewCar2 receives message, starts reducing speed with -0.5m/s² during ToC interval. The following HMI is shown:

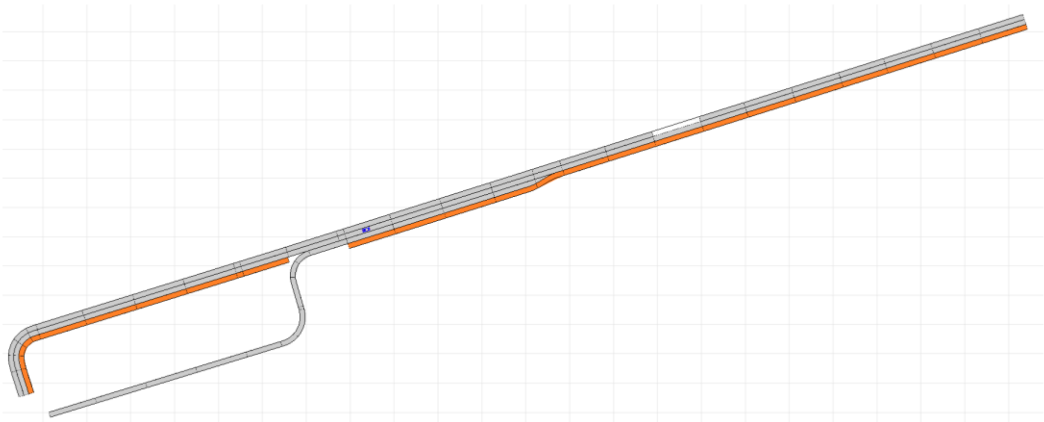
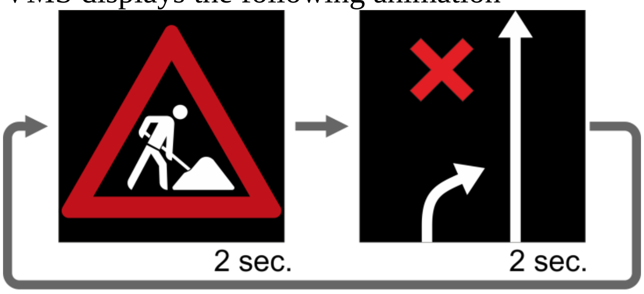
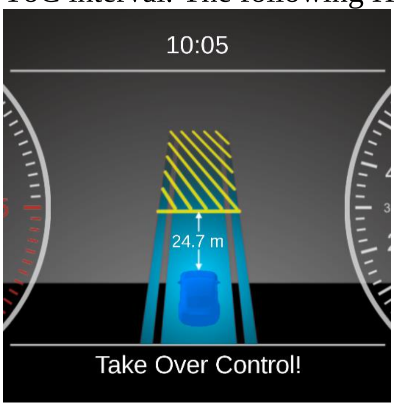
<table>
<tr><td></td><td>

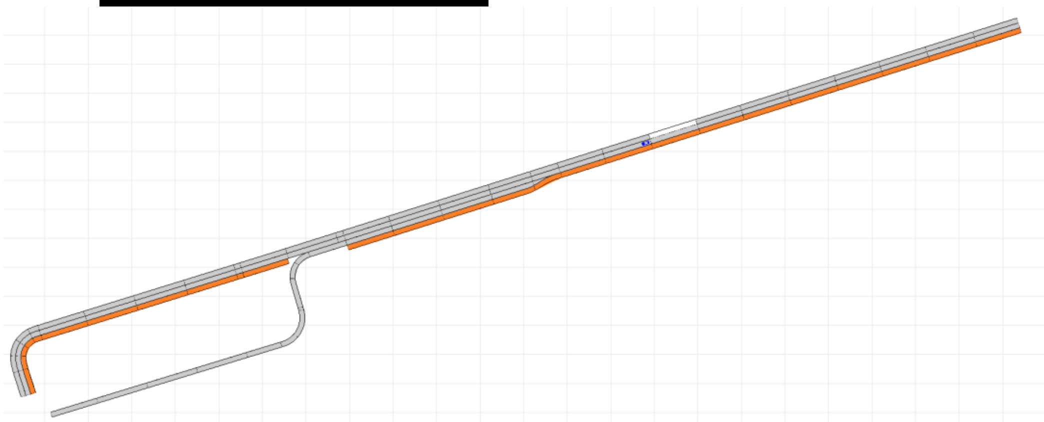

5. Driver ignores ToC
6. RSU sends MCM:: RSU_SMC ::LaneAdvice=3(=SafeSpot position) to ViewCar2
7. MRM is executed. The following HMI is shown:


8. ViewCar2 performs lane change and stops in the SafeSpot,

</td></tr>
<tr><td>Associated to Requirement(s)</td><td>
• REQ_V_I1_S4.2_1
• REQ_V_I1_S4.2_2
• REQ_I_I1_S4.2_1
• REQ_I_I1_S4.2_2
</td></tr>
</table>

3.2.5.2.3 Test 4.2_2: “MRM on current lane, SafeSpot occupied”

Goal	Demonstrate infrastructure behaviour in case of an occupied safe spot. Minimum Risk Maneuver is performed on the driving lane.

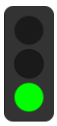
Used vehicles	ViewCar2
Used infrastructure	RSU, Camera, VMS
Used messages	CAM, DENM, I2V-MCM
Initial situation	ViewCar2 starts on right lane of two-lane road, followed by a legacy vehicle. The left lane of the road is blocked by roadworks. The vehicle is by default not able to pass the roadworks on the right lane. 
Scenario script	1. VMS displays the following animation  2. RSU broadcasts DENM::Roadworks 3. RSU sends MCM:: RSU_SMC ::ToCAdvice to ViewCar2 4. ViewCar2 receives message, starts reducing speed with -0.5m/s² during ToC interval. The following HMI is shown:  5. Driver ignores ToC 6. MRM is executed, ViewCar2 stops in right lane. The following HMI is shown:

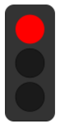
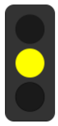
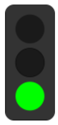
	 
Associated to Requirement(s)	• REQ_V_I1_S4.2_1 • REQ_V_I1_S4.2_2 • REQ_I_I1_S4.2_1 • REQ_I_I1_S4.2_2

3.2.5.3 Feasibility results

3.2.5.3.1 Requirements verification

In addition to the feasibility assessment of the general requirements shown in section 3.2.1 a few service-specific requirements needed to be verified:

Service-specific requirement description		Req. Name	Associated Test cases successfully executed	Notes
Vehicle Requirements	*Safe spot advice following* The CAVs need to be able to receive safe spot advices from the infrastructure. The advices need to be taken into account during trajectory and Minimum Risk Manoeuvre planning. It may be necessary, that the current level of automation is also communicated to the infrastructure.	REQ_V_I1_S4.2_1		Safe spot advices received and followed using the lane change and ToC advices available in the MCM´s RSU container.

	Manoeuvre Coordination Message support The vehicles need to provide manoeuvre information in order to be able to implicitly block safe spots. Manoeuvres of the other vehicles shall be received and taken into account for the own trajectory and Minimum Risk Manoeuvre planning.	REQ_V_I1_S4.2_2		MCM provided, but MCM-V2V support only tested in simulation.
Infrastructure requirements	*Safe spot availability detection* The infrastructure needs the capability to always track the availability of the safe spots. This does not only include listening to appropriate messages indicating the blockage, but also the detection by using e.g. camera systems. This is necessary, as the safe spot areas may be also blocked by non-cooperative vehicles, e.g. due to a brake-down of a legacy vehicle.	REQ_I_I1_S4.2_1		Safe spot availability was followed by using MCM. Nevertheless, the safe spot availability was not detected online by camera or message reception.
	Safe spot advice generation Whenever a safe spot is available, the infrastructure should forward this information to the vehicles.	REQ_I_I1_S4.2_2		Safe spot advice was provided by RSU to receiving vehicles.

For this set of tests, the reception and transmission of required V2X message was also verified using the V2X module (CohdaWireless mk5) present on the test track. The capture log shows that the RSU correctly formats DENM and MCM messages, and the content of these messages fits to the specific requirements of the tests under evaluation. In particular, the DENM shows the event position of the roadworks and the lanes that are closed, and the MCM includes the ToC Advice and Lane Advice when required. The safe spot information to perform the MRM (test 4.2_1) is indicated by making the place of end transition to match the lane change position, so that if the driver does not take control, the MRM coincides with the lane change. Safe spot information will be an extension of the MCM message for the TransAID´s second iteration.

3.2.5.3.2 User experience

MRMs were successfully executed during the test cases. Deceleration speed was still acceptable from user’s point of view. A potential step before starting the MRM could be a minimal steering jerk and/or activation of the vehicles break system to trigger the driver’s attention that a vehicle control takeover is requested to reduce the chance a MRM must be triggered.

3.2.5.3.3 Check of overall feasibility

The bad impact of MRM was successfully demonstrated, which also leads to the conclusion that it is recommended to introduce safe spots (in areas where it is feasible, cf. road architecture). Additional space (safe spots) in front of road works could also have positive side-effects on the safety level of road workers: In case of accidents a safe spot can reduce the impact of vehicle accidents (speed mitigation before hitting objects of the road works).

3.2.6 Requirements of Scenario 5.1: Schedule ToCs before no AD zone

3.2.6.1 Description of scenario from D2.2

After a transition of control (ToC) from automated to manual mode, an automated vehicle is expected to behave more erratically. The driving characteristics are different (e.g. different headway, different lateral movement variation, different overtaking behaviour, etc.). Because the driving behaviour during transitions and driving behaviour shortly thereafter are different, traffic flow and safety are disturbed. This effect is amplified when there are many ToCs in the same area. To prevent that amplification in mixed traffic scenarios, downward ToCs are distributed in time and space upstream of an area where there is no or limited automated driving (e.g. tunnel, geofence, complicated road works).

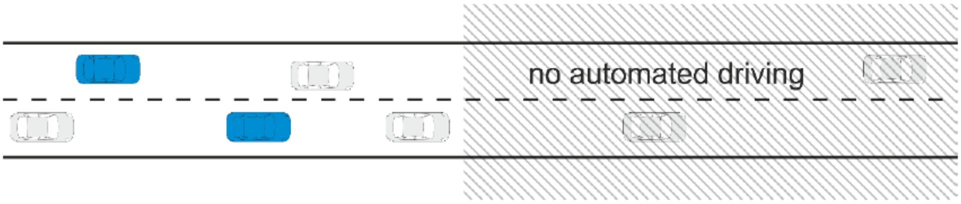


Figure 29: schematic overview of Scenario 5.1

Figure 29 shows the Scenario 5.1 where CAVs and other traffic are approaching a no AD zone with 2 lanes. Starting at some point upstream of the no AD zone, the RSI determines through collective perception the positions and speeds of vehicles and determines the optimal location and moment for CAVs to perform a downward ToC. Subsequently, ToC requests are provided to the corresponding CAVs. Based on the ToC requests, the CAVs perform ToCs at the desired location and moment in time. CVs are warned about the ToCs and possible MRMs. In the no AD zone, the CAVs are in manual mode.

Note: the figure is schematic. The blue automated vehicles have performed ToCs further upstream than the picture might suggest.

3.2.6.2 Scenario setup

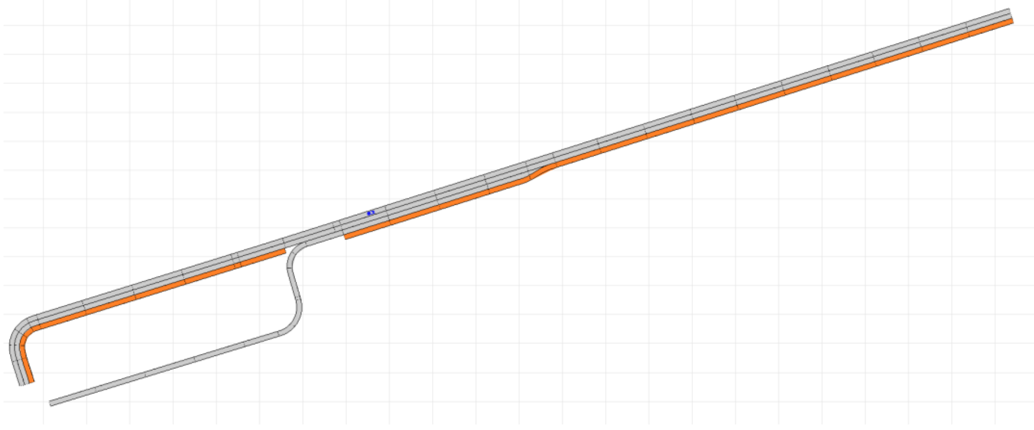
The effects of this scenario can best be seen in traffic simulation. Nevertheless, the feasibility should be shown as well. Therefore, ToC advice messages need to be implemented and tested. If the infrastructure needs more information to trigger the ToC advice messages, the scenario can be extended accordingly.

Figure 30: DLR's ViewCar2 executing the scenario 5.1 tests.

For scenario 5.1, two tests are performed, described in the following.

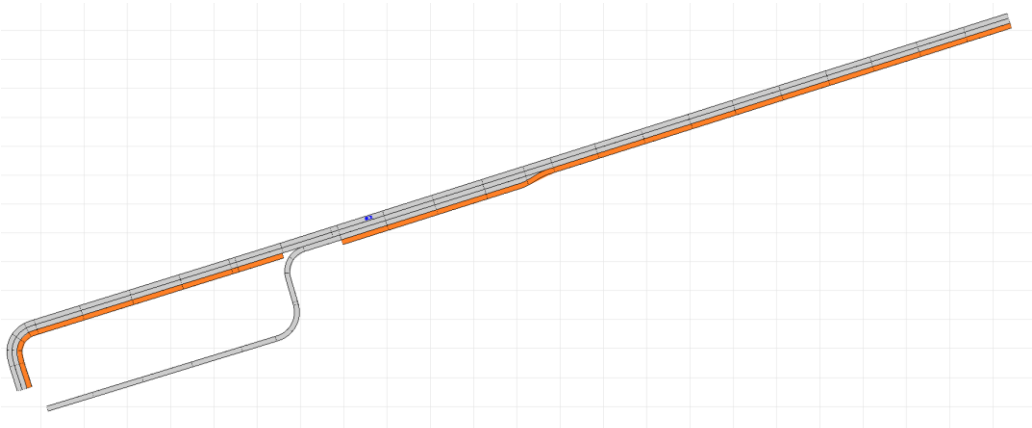
3.2.6.2.1 Test 5.1_0: "Scheduled ToCs with driver's response"

Goal	Demonstrate the possibility of scheduled ToCs. In this case, the driver is responding and the ToC is successful.
Used vehicles	ViewCar2
Used infrastructure	RSU, VMS
Used messages	CAM, DENM, I2V-MCM
Initial situation	ViewCar2 starts on right lane of two-lane road. It approaches an area where no automated driving is possible or allowed

	
Scenario script	1. VMS displays the following sign:  2. RSU optionally broadcasts DENM::NoADZone 3. RSU sends MCM:: RSU_SMC ::ToCAdvice (400m behind reference point) to ViewCar2 4. ViewCar2 issues ToC. The following HMI is shown:  5. Driver takes over control.
Associated to Requirement(s)	• REQ_V_I1_S5.1_1 • REQ_I_I1_S5.1_1

3.2.6.2.2 Test 5.1_1: "Scheduled ToCs without driver's response"

Goal	Demonstrate the possibility of scheduled ToCs. In this case, the driver is not responding and the ToC is unsuccessful.
Used vehicles	ViewCar2
Used infrastructure	RSU, VMS

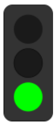
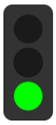
Used messages	CAM, DENM, I2V-MCM
Initial situation	ViewCar2 starts on right lane of two-lane road. It approaches an area where no automated driving is possible or allowed 
Scenario script	1. VMS displays the following sign:  2. RSU optionally broadcasts DENM::NoADZone 3. RSU sends MCM:: RSU_SMC ::ToCAdvice (400m behind reference point) to ViewCar2 4. ViewCar2 issues ToC. The following HMI is shown:  5. Driver does not take over control, MRM is executed. ViewCar2 stops on current lane. The following HMI is shown:

	
Associated to Requirement(s)	• REQ_V_I1_S5.1_1 • REQ_I_I1_S5.1_1

3.2.6.3 Feasibility results

3.2.6.3.1 Requirements verification

In addition to the feasibility assessment of the general requirements shown in section 3.2.1 a few service-specific requirements needed to be verified:

Service-specific requirement description		Req. Name	Associated Test cases successfully executed	Notes
Vehicle requirements	*ToC advice following* The CAVs need to be able to receive ToC advice from the infrastructure. The advice needs to be taken into account while driving. It may be necessary, that the current level of automation is also communicated to the infrastructure.	REQ_V_I1_S5.1_1		ToC advice received and followed. The vehicles report the current level of automation to the infrastructure using an extended CAM container. This was not implemented by design for the TransAID first iteration.
Infrastructure requirements	*ToC advice generation* The infrastructure needs to be able to generate ToC advice. The exact requirements for this need to be derived from the baseline simulations and the envisioned traffic management procedures.	REQ_I_I1_S5.1_1		ToC advice generated by present RSU.

For this set of tests, the reception and transmission of required V2X messages was also verified using the V2X module (CohdaWireless mk5) present on the test track. The capture log shows that the RSU correctly formats MCM messages, and that the content of these messages fits to the specific requirements of the tests under evaluation. In particular, the MCM includes two ToC advice entries. The advice is addressed to two different stations, and they indicate the place where the transition of control should be completed before executing the MRM.

3.2.6.3.2 User experience

A passenger of a CAV could not identify that the behaviour changed here (scheduling of ToC) compared to a fixed time or spot where the ToC is triggered (cf. Test 1.1_0: “Baseline: ToC in front of blockage”). Results and comments from previous ToC related scenarios also apply here. A proper HMI will have a high influence on the level of comfort and the perceived safety.

3.2.6.3.3 Check of overall feasibility

Feasibility of a scheduled ToC is expected to reduce the chance of stopped CAV/CV or generation of traffic jams. This was not verified in this first stage implementation due to the lack of test vehicles.

4 Conclusion and outlook

This deliverable described the implementations done during the first project iteration for the real world prototype. This prototype consists of cooperative automated vehicles and different infrastructure components, as well as implementations of the message sets defined in D5.1.

The feasibility of the real-world implementation of both the developed message sets and traffic management measures of the first project iteration has been shown for all TransAID services and scenarios. Everything can be put into real world vehicles and infrastructure to allow future traffic systems to run more stable and with fewer problems for automated vehicles by using infrastructure-based advice.

Nevertheless, some minor issues with some of the fields of the used messages have been found, which gives valuable input for the second project iteration. Also, some minor shortcomings in the different software implementations could be identified, esp. in the field of stability and complexity. The software components will therefore be further improved and enhanced during the second iteration, so that the prototypes are also ready to be shown in public traffic conditions.

References

[1] J. Schindler, R. Markowski, F. Andert, A. Correa, M. Rondinone und R. Blokpoel, „TransAID Deliverable D7.1: System architecture for real world vehicles and road side,“ 2019.

[2] A. Wijbenga, E. Mintsis, J. Vreeswijk, A. Correa, L. Lücken, J. Schindler, M. Rondinone, S. Maerivoet, L. Akkermans, K. Carlier, I. Mayeres, E. Mitsakis, M. Sepulcre und R. Markowski, „TransAID Deliverable D2.2: Scenario definitions and modelling requirements,“ 2018.

[3] M. Rondinone, A. Correa, R. Blokpoel, F. Andert, S. Khan, A. Wijbenga, J. Vreeswijk, M. Sepulcre und J. Gozalvez, „TransAID Deliverable D5.1: Definition of V2X message sets,“ 2018.

[4] A. Correa, F. Andert, R. Blokpoel, N. Wojke, G. Thandavarayan, B. Coll, M. Sepulcre, J. Gozalvez, A. Wijbenga, J. Schindler und S. Maerivoet, „TransAID Deliverable D5.2: V2X-based cooperative sensing and driving in Transition Areas,“ 2019.

[5] S. Maerivoet, L. Akkermans, K. Carlier, Y.-P. Flötteröd, L. Lücken, R. Alms, E. Mintsis, D. Koutras, A. Wijbenga, J. Vreeswijk, A. Correa, X. Zhang und R. Blokpoel, „TransAID Deliverable D4.2: Preliminary simulation and assessment of enhanced traffic management measures,“ 2018.

[6] L. Lücken, M. Schwamborn, E. Mintsis, D. Koutras, A. Correa, M. Sepulcre, B. Coll, R. Blokpoel, G. Huisken, S. Maerivoet und J. Schindler, „TransAID Deliverable D6.2: Assessment of Traffic Management Procedures in Transition Areas,“ 2019.

[7] M. Fischer, A. Richter, J. Schindler, J. Plättner, G. Temme, J. Kelsch, D. Assmann und F. Köster, „Modular and Scalable Driving Simulator Hardware and Software for the Development of Future Driver Assistance and Automation Systems,“ in *New Developments in Driving Simulation Design and Experiments*, Paris, 2014.

[8] D. Heß, S. Lapoehn, F. Utesch, M. Fischer, J. Schindler, T. Hesse und F. Köster, „Contributions of the EU Projects UnCoVerCPS and Enable-S3 to Highly Automated Driving in Conflict Situations,“ *AAET Automatisiertes und Vernetztes Fahren,* February 2019.

[9] D. Heß, R. Lattarulo, P. Joshue, J. Schindler, T. Hesse und F. Köster, „Fast Maneuver Planning for Cooperative Automated Vehicles,“ *21st IEEE International Conference on Transportation Systems (ITSC),* October 2018.

[10] H. J. Ferreau, C. Kirsches, A. Potschka, H. G. Bock und M. Diehl, „qpOASES: A parametric active-set algorithm for quadratic programming.,“ in *Mathematical Programming Computation*, 4 Hrsg., Bd. 6, 2014, pp. 327-363.

[11] *ETSI EN 302 637-2 Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service, V1.3.2*, 2014.

[12] *ETSI EN 302 665 Intelligent Transport Systems (ITS) Communication Architecture, v1.1.1,* 2010.

[13] *ETSI EN 302 637-3 Intelligent Transport Systems (ITS) Vehicular Communications: Basic Set of Applications; Part 3: Specification of Decentralized Environmental Notification Basic Service, v1.2.2,* 2014.

[14] *ETSI TS 102 894-2 Intelligent Transport Systems (ITS); Users and applications requirements; Part 2: Applications and facilities layer common data dictionary V1.2.1,* 2014.

[15] *ETSI TS 102 894-1 Intelligent Transport Systems (ITS); Users and application requirements; Part 1: Facility layer structure, functional requirements and specifications v1.1.1*, 2013.

[16] *ETSI EN 302 931 Intelligent Transport Systems (ITS) Vehicular Communications; Geographical Area Definition, v1.1.1,* 2011.

[17] *ETSI EN 302 636-5-1 Intelligent Transport Systems (ITS) Vehicular Communications; GeoNetworking; Part 5: Transport protocols; Sub-part 1: Basic Transport Protocol, v1.2.1,* 2014.

[18] *ETSI TS 103 248 Intelligent Transport Systems (ITS) GeoNetworking; Port Numbers for the Basic Transport Protocol (BTP), v1.0.1,* 2016.

[19] *ETSI TS 103 301 Intelligent Transport Systems (ITS) Vehicular Communications; Basic Set of Applications; Facilities layer protocols and communication requirements for infrastructure services, v1.1.1,* 2016.

[20] *ETSI TS 102 63-4-2 ITS Vehicular Communications; GeoNetworking; Part 4: Geographical addressing and forwarding for point-to-point and point-to-multipoint communications; Sub-part 2: Media-dependent functionalities for ITS-G5, v1.1.1,* 2013.

[21] *ETSI ES 202 663 Intelligent Transport Systems (ITS); European profile standard for the physical and medium access control layer of Intelligent Transport Systems operating in the 5 GHz frequency band, V1.1.0,* 2010.

[22] N. Wojke, A. Bewley und D. Paulus, „Simple online and realtime tracking with a deep association metric,“ *IEEE International Conference on Image Processing (ICIP),* pp. 3645-3649, September 2017.

[23] A. Wijbenga, E. Mintsis, J. Schindler, J. Vreeswijk, M. Rondinone, A. Correa, M. Sepulcre und S. Maerivoet, „TransAID Deliverable D2.1: Use cases and safety and efficiency metrics,“ 2018.

[24] M. Rondinone, R. Blokpoel, J. Vreeswijk und J. Schindler, „MAVEN Deliverable D5.1: V2X communications for infrastructure-assisted automated driving,“ 2018.

Annex A: TransAID messages description

Annex A1: MCM description

			Container		Data Frame		Opt.	Type	Description
MCM	ITS PDU Header				Protocol version			Integer 0-255	version of the ITS message and/or communication protocol
					Message id			Integer 0-255	Type of the ITS message
					Station id			Integer 0-4294967295	Identifier for an ITS-S
	Maneuver Coordination	Generation Delta Time						Integer 0-65535	Time corresponding to the time of the reference position in the MCM, considered as time of the CPM generation. The value of the DE shall be wrapped to 65 536. This value shall be set as the remainder of the corresponding value of TimestampIts divided by 65 536 as below: generationDeltaTime = TimestampIts mod 65 536
		MCM Parameters	Basic Container	Station type				Integer 0-255	The type of technical context the ITS-S is integrated in. The station type depends on the integration environment of ITS-S into vehicle, mobile devices or at infrastructure. Types: unknown(0), pedestrian(1), cyclist(2), moped(3), motorcycle(4), passengerCar(5), bus(6), lightTruck(7), heavyTruck(8), trailer(9), specialVehicles(10), tram(11), roadSideUnit(15)
				Reference position	Latitude			Integer 900000000-900000001	Latitude: oneMicrodegreeNorth(10), oneMicrodegreeSouth(-10), unavailable(900000001)
					Longitude			Integer -1800000000..1800000001	Longitude: oneMicrodegreeEast(10), oneMicrodegreeWest(-10), unavailable(1800000001)
					Position Confidence Ellipse			Sequence of semiMajorConfidence, semiMinorConfidence, semiMajorOrientation	The positionConfidenceEllipse provides the accuracy of the measured position with the 95 % confidence level. Otherwise, the positionConfidenceEllipse shall be set to unavailable.If semiMajorOrientation is set to 0° North, then the semiMajorConfidence corresponds to the position accuracy in the North/South direction, while the semiMinorConfidence corresponds to the position accuracy in the East/West direction. This definition implies that the semiMajorConfidence might be smaller than the semiMinorConfidence.
					Altitude	Value		Integer -100000-800001	Altitude: referenceEllipsoidSurface(0), oneCentimeter(1), unavailable(800001)
						Conf		Enumerated 0-15	alt-000-01(0), alt-000-02(1), alt-000-05(2), alt-000-10(3), alt-000-20(4), alt-000-50(5), alt-001-00(6), alt-002-00(7), alt-005-00(8), alt-010-00(9), alt-020-00(10), alt-050-00(11), alt-100-00(12), alt-200-00(13), outOfRange(14), unavailable(15)
			Maneuver Container					Choice	Choice between Vehicle Maneuver Container or RSU Suggested Maneuver containders
			Vehicle Maneuver Container		Tolerated Distance Ahead			Integer 0-10000	The tolerated distance is the distance to the trajectory points that other vehicles have to respect when they want to accept a desired trajectory of someone else
					Tolerated Distance Behind			Integer 0-10000	The tolerated distance is the distance to the trajectory points that other vehicles have to respect when they want to accept a desired trajectory of someone else
					Planned Trajectory			Sequence size 1-30 of Trajectory Points	Future trajectory of the vehicle
					Trajectory Points	deltaXcm		Integer 0-10000	The trajectory points are composed by delta-values in the vehicle coordinate system The reference position of the first point is the position and heading stated in the MCM Each following position (n) references to the former position (n-1)
						deltaYcm		Integer 0-10000	

	Container	Data Frame			Opt.	Type	Description
				deltaTimeMs		Integer 0-65535	
				absSpeed	X	Integer 0-16383	SpeedValue: standstill(0), oneCentimeterPerSec(1), unavailable(16383)
				Longitudinal acceleration	X	Integer -160-161	LongitudinalAccelerationValue: pointOneMeterPerSecSquaredForward(1), pointOneMeterPerSecSquaredBackward(-1), unavailable(161)
		Desired Trajectory			X	Sequence size 1-30 of Trajectory Points	Desired trajectory if other vehicles agree
			Trajectory Points	deltaXcm		Integer 0-10000	The trajectory points are composed by delta-values in the vehicle coordinate system The reference position of the first point is the position and heading stated in the MCM Each following position (n) references to the former position (n-1)
				deltaYcm		Integer 0-10000	
				deltaTimeMs		Integer 0-65535	
				absSpeed		Integer 0-16383	SpeedValue: standstill(0), oneCentimeterPerSec(1), unavailable(16383)
				Longitudinal acceleration		Integer -160-161	LongitudinalAccelerationValue: pointOneMeterPerSecSquaredForward(1), pointOneMeterPerSecSquaredBackward(-1), unavailable(161)
		RespectedDesiredTrajectoriesList				Sequence size 0-5 of RespectedDesiredTrajectory	
			RespectedDesiredTrajectory			Integer 0-4294967295	Reflects **the** vehicle ID which is respected in planning
		TriggerTimeOfToC			X		Time when the ToC process starts
				Minute		Integer 0-527040	Time when the ToC will be triggered in minutes since the start of the year
				Milisecond		Integer 0-65535	Time when the ToC will be triggered in milicsecons since the start of the minute
		TargetAutomationLevel			X	Enumerated	Level of automation of the vehicle after the ToC: saeLevel0 (0), saeLevel1 (1), saeLevel2 (2), saeLevel3 (3), saeLevel4 (4), saeLevel5 (5), ...
		TriggerTimeOfMRM			X	Integer 0-65535	Time in miliseconds since the trigger of the ToC when the MRM will be triggered if the driver does not take control of the car
		Heading					
				Value		Integer 0-3601	Orientation of a heading with regards to the WGS84 north: wgs84North(0), wgs84East(900), wgs84South(1800), wgs84West(2700), unavailable(3601)
				Confidence		Integer 0-127	The absolute accuracy of a reported heading value for a predefined confidence level: equalOrWithinZeroPointOneDegree(1), equalOrWithinOneDegree(10), outOfRange(126), unavailable(127)
		Speed					
				Value		Integer 0-16383	speed value: standstill(0), oneCentimeterPerSec(1), unavailable(16383)
				Confidence		Integer 0-127	The absolute accuracy of a reported speed value for a predefined confidence level: equalOrWithinOneCentimeterPerSec(1), equalOrWithinOneMeterPerSec(100), outOfRange(126), unavailable(127)
		Longitudinal Acceleration					
				Value		Integer -160-161	Vehicle acceleration at longitudinal direction in the centre of the mass of the empty vehicle
				Confidence		Integer 0-102	The absolute accuracy of a reported acceleration value for a predefined confidence level: pointOneMeterPerSecSquaredForward(1), pointOneMeterPerSecSquaredBackward(-1), unavailable(161)
		Lateral acceleration					
				Value		Integer -160-161	Vehicle acceleration at lateral direction in the centre of the mass of the empty vehicle: pointOneMeterPerSecSquaredToRight(-1), pointOneMeterPerSecSquaredToLeft(1), unavailable(161)

			Container	Data Frame			Opt.	Type	Description
						Confidence		Integer 0-102	The absolute accuracy of a reported acceleration value for a predefined confidence level: pointOneMeterPerSecSquared(1), outOfRange(101), asfaunavailable(102)
				Vertical Acceleration			X		
						Value		Integer -160-161	Vehicle acceleration at vertical direction in the centre of the mass of the empty vehicle: pointOneMeterPerSecSquaredUp(1), pointOneMeterPerSecSquaredDown(-1), unavailable(161)
						Confidence		Integer 0-102	The absolute accuracy of a reported acceleration value for a predefined confidence level: pointOneMeterPerSecSquared(1), outOfRange(101), unavailable(102)
				Yaw Rate					
						Value		Integer -32766 - 32767	Vehicle rotation around z-axis of coordinate system centred on the centre of mass of the empty-loaded vehicle: straight(0), degSec-000-01ToRight(-1), degSec-000-01ToLeft(1), unavailable(32767)
						Confidence		Enumerated	The absolute accuracy range for reported yaw rate value for a predefined confidence level: degSec-000-01(0), degSec-000-05(1), degSec-000-10(2), degSec-001-00(3), degSec-005-00(4), degSec-010-00(5), degSec-100-00(6), outOfRange(7), unavailable(8)
				Curvature					
						Value		Integer -30000 - 30001	The inverse of a detected vehicle turning curve radius scaled with 30 000A curvature detected by a vehicle represents the curvature of the actual vehicle trajectory: straight(0), reciprocalOf1MeterRadiusToRight(-30000), reciprocalOf1MeterRadiusToLeft(30000), unavailable(30001)
						Confidence		Enumerated	The absolute accuracy range of a reported curvature value for a predefined confidence level: onePerMeter-0-00002(0), onePerMeter-0-0001(1), onePerMeter-0-0005(2), onePerMeter-0-002(3), onePerMeter-0-01(4), onePerMeter-0-1(5), outOfRange(6), unavailable(7)
				Curvature Calculation Mode				Enumerated	It describes whether the yaw rate is used to calculate the curvature for a reported curvature value: yawRateUsed(0), yawRateNotUsed(1), unavailable(2), ...
				Drive Direction				Enumerated	It denotes whether a vehicle is driving forward or backward: forward(0), backward(1), unavailable(2)
				Lane Position				Integer -1 - 14	the transversal position information on the road in resolution of lanes, counted from the outside border of the road for a given traffic direction: offTheRoad(-1), hardShoulder(0), outermostDrivingLane(1), secondLaneFromOutside(2)
				Steering Wheel Angle					
						Value		Integer -511 - 512	Steering wheel angle of the vehicle at certain point in time: straight(0), onePointFiveDegreesToRight(-1), onePointFiveDegreesToLeft(1), unavailable(512)
						Confidence		Integer 1 - 127	The Absolute accuracy for a reported steering wheel angle value for a predefined confidence level: equalOrWithinOnePointFiveDegree(1), outOfRange(126), unavailable(127)
				Advice Response List			X	Sequence size 0-3 of Advice Response	List of advice responce objects
					Advice Response	Advice ID		Integer 0-255	Identifier for the ackowledgement
						Advice Followed		Bit String	Advice response: 1 followed, 0 not followed
			RSU Suggested	IntersectionReferenceID			X		Specific lane ids are referring to this

Container	Data Frame				Opt.	Type	Description
Maneuver Container							intersection id
		RoadRegulatorID			X	Integer 0-65535	A globally unique regional assignment value typical assigned to a regional DOT authority. The value zero shall be used for testing needs
		IntersectionID				Integer 0-65535	A unique mapping to the intersection in question within the above region of use
	RoadSegmentReferenceID				X		Specific lane ids are referring to this roadsegment id
		RoadRegulatorID			X	Integer 0-65535	A globally unique regional assignment value typical assigned to a regional DOT authority. The value zero shall be used for testing needs
		RoadSegmentID				Integer 0-65535	A unique mapping to the road segment in question within the above region of use during its period of assignment and use. Note that unlike intersectionID values, this value can be reused by the region
	VehicleAdviceList				X	Sequence of size 8 of Vehicle Advice	List of lane advice objects, one per vehicle
	Vehicle Advice	Target Station ID				Integer 0-4294967295	StationID of the vehicle the advice is targeted at
		Lane Advice			X		Single lane advice object
			Advice ID			Integer 0-255	Identifier for acknowledgement
			LaneAdviceReason			Enumerated	Indicates the reason why the CAV should perform the lane change: reason0 (0), reason1 (1), ...
			LaneChangePosition			Integer 0-10000	Position where the lane change should take place
			LaneChangeMoment				Time when the lane change should be performed
				Minute		Integer 0 - 527040	Time when the lane change should start in minutes since the start of the year
				Milllisecond		Integer 0 - 65535	Time when the lane chage should start in milicsecons since the start of the minute
			LaneChangeSpeed		X	Integer 0-500	Speed advice at the moment of the lane change
			LeadingVehicle		X	Integer 0-4294967295	StationID of the vehicle intented to be ahead of the target vehicle after merging
			FollowingVehicle		X	Integer 0-4294967295	StationID of the vehicle intented to be behind of the target vehicle after merging
			TargetLane			Integer 0-255	The lane number towards the target vehicle should move
			TriggeringPointOfToC		X	Integer 0-10000	Distance from the starting point where a ToC should be triggered if the lane change is not performed
		Car Following Advice			X		Single speed advice object
			Advice ID			Integer 0-255	Identifier for acknowledgement
			AdviceLaneID			Integer 0-255	LaneID to which the advice and position applies
			AdvicePosition			Integer 0-10000	Position where the target speed should be adhered
			DesiredBehaviour			Choice	Choice between TargetGap and TartetSpeed
				TargetGap		Integer 0-255	Target distance in m towards vehicle ahead
				TargetSpeed		Integer 0-255	Value of the speed advised to the target vehicle
		ToC Advice			X		
			Advice ID			Integer 0-255	Identifier for acknowledgement
			TocAdviceReason			Enumerated	Indicates the reason why the CAV should perform the ToC: reason0 (0), reason1 (1), ...
			PlaceOfStartTransition		X	Integer 0-10000	Position where the ToC should start
			TimeOfTriggerTransition		X		Time when the ToC should start
				Minute		Integer 0 - 527040	Time when the ToC should start in minutes since the start of the year
				Millisecond		Integer 0 - 65535	Time when the ToC should start in miliseconds since the start of the minute
			PlaceOfEndTransition		X	Integer 0-10000	Distance from the starting point where the ToC can be done

Annex B: TransAID messages ASN.1 specifications

Annex B1: MCM ASN.1 specification

```
MCM-TransAID DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

IMPORTS
ItsPduHeader, StationType, ReferencePosition, Heading, Speed,
LongitudinalAcceleration, LateralAcceleration, VerticalAcceleration, YawRate,
Curvature, CurvatureCalculationMode,DriveDirection, LanePosition,
SteeringWheelAngle, SpeedValue, LongitudinalAccelerationValue

FROM ITS-Container {itu-t(0) identified-organization(4) etsi(0) itsDomain(5)
wg1(1) ts(102894) cdd(2) version(1)};

MCM ::= SEQUENCE {
        header                  ItsPduHeader,
        maneuverCoordination    ManeuverCoordination
}

ManeuverCoordination ::= SEQUENCE {
        generationDeltaTime     GenerationDeltaTime,
        mcmParameters           McmParameters
}

GenerationDeltaTime ::= INTEGER {
        oneMilliSec(1)
} (0..65535)

McmParameters ::= SEQUENCE {
        basicContainer          BasicContainer,
        maneuverContainer       ManeuverContainer
}

ManeuverContainer ::= CHOICE {
        vehicleManeuver         VehicleManeuver,
        rsuManeuver             RsuManeuver,
        ...
        }

BasicContainer ::= SEQUENCE {
        stationType             StationType,
        referencePosition       ReferencePosition,
        ...
}

VehicleManeuver ::= SEQUENCE {
        toleratedDistanceAheadCmps     ToleratedDistance,
        toleratedDistanceBehindCmps    ToleratedDistance,
        plannedTrajectory              PlannedTrajectory,
        desiredTrajectory              DesiredTrajectory OPTIONAL,
        respectedDesiredTrajectoriesList RespectedDesiredTrajectoriesList,
        triggerTimeOfToC               TriggerTimeOfToC OPTIONAL,
        targetAutomationLevel          TargetAutomationLevel OPTIONAL,
        triggerTimeOfMRM               TriggerTimeOfMRM OPTIONAL,
        heading                        Heading,
```

```
        speed                          Speed,
        longitudinalAcceleration       LongitudinalAcceleration,
        lateralAcceleration            LateralAcceleration,
        verticalAcceleration           VerticalAcceleration,
        yawRate                        YawRate,
        curvature                      Curvature,
        curvatureCalculationMode       CurvatureCalculationMode,
        driveDirection                 DriveDirection,
        lanePosition                   LanePosition,
        steeringWheelAngle             SteeringWheelAngle,
        adviceResponseList             AdviceResponseList OPTIONAL
}

ToleratedDistance ::= INTEGER (0..10000)

PlannedTrajectory ::= SEQUENCE SIZE (1..30) OF TrajectoryPoint

DesiredTrajectory ::= SEQUENCE SIZE (1..30) OF TrajectoryPoint

TrajectoryPoint ::= SEQUENCE {
        deltaXCm                       DiffPosition,
        deltaYCm                       DiffPosition,
        deltaTimeMs                    DiffTime,
        absSpeed                       SpeedValue OPTIONAL,
        longitudinalAcceleration       LongitudinalAccelerationValue OPTIONAL
}

DiffPosition ::= INTEGER (0..10000)

DiffTime ::= INTEGER (0..65535)

RespectedDesiredTrajectoriesList ::= SEQUENCE SIZE (0..5) OF
RespectedDesiredTrajectory

RespectedDesiredTrajectory ::= INTEGER (0..4294967295)

TriggerTimeOfToC ::= SEQUENCE {
        minute                 Minute,
        millisecond            Millisecond
}

Minute ::= INTEGER (0..527040)

Millisecond ::= INTEGER (0..65535)

TargetAutomationLevel ::= ENUMERATED {
        saeLevel0 (0),
        saeLevel1 (1),
        saeLevel2 (2),
        saeLevel3 (3),
        saeLevel4 (4),
        saeLevel5 (5),
        ...
}

TriggerTimeOfMRM ::= INTEGER (0..65535)
```

```
AdviceResponseList ::= SEQUENCE SIZE (0..3) {
        adviceResponse         AdviceResponse
}

AdviceResponse ::= SEQUENCE {
        adviceID               AdviceID,
        adviceFollowed AdviceFollowed
}

AdviceID ::= INTEGER (0..255)

AdviceFollowed ::= BIT STRING {
        notFollowed(0),
        followed(1)
}

RsuManeuver ::= SEQUENCE {
        intersectionReferenceID       IntersectionReferenceID OPTIONAL,
        roadSegmentReferenceID        RoadSegmentReferenceID OPTIONAL,
        vehicleAdviceList             VehicleAdviceList OPTIONAL
}

IntersectionReferenceID ::= SEQUENCE {
        region  RoadRegulatorID OPTIONAL,
        id IntersectionID
}

RoadSegmentReferenceID ::= SEQUENCE {
        region RoadRegulatorID OPTIONAL,
        id     RoadSegmentID
}

RoadRegulatorID ::= INTEGER (0..65535)

IntersectionID ::= INTEGER (0..65535)

RoadSegmentID ::= INTEGER (0..65535)

VehicleAdviceList ::= SEQUENCE SIZE (1..8) OF VehicleAdvice

VehicleAdvice ::= SEQUENCE {
        targetStationID        TargetStationID,
        laneAdvice             LaneAdvice OPTIONAL,
        carFollowingAdvice     CarFollowingAdvice OPTIONAL,
        tocAdvice              TocAdvice OPTIONAL
}

TargetStationID ::= INTEGER (0..4294967295)
```

```
LaneAdvice ::= SEQUENCE {
        adviceID                AdviceID,
        laneAdviceReason        LaneAdviceReason,
        laneChangePosition      LaneChangePosition,
        laneChangeMoment        LaneChangeMoment,
        laneChangeSpeed         LaneChangeSpeed OPTIONAL,
        leadingVehicle          LeadingVehicle OPTIONAL,
        followingVehicle        FollowingVehicle OPTIONAL,
        targetLane              TargetLane,
        triggeringPointOfToC    TriggeringPointOfToC OPTIONAL
}

CarFollowingAdvice ::= SEQUENCE {
        adviceID                AdviceID,
        adviceLaneID            AdviceLaneID,
        advicePosition          AdvicePosition,
        desiredBehaviour        DesiredBehaviour
}

TocAdvice ::= SEQUENCE {
        adviceID                AdviceID,
        tocAdviceReason         TocAdviceReason,
        placeOfStartTransition PlaceOfStartTransition OPTIONAL,
        timeOfTriggerTransition TimeOfTriggerTransition OPTIONAL,
        placeOfEndTransition   PlaceOfEndTransition OPTIONAL
}

RequestID ::= INTEGER (0..255)

LaneAdviceReason ::= ENUMERATED {
        reason0 (0),
        reason1 (1),
        ...
}

LaneChangePosition ::= INTEGER (0..10000)

LaneChangeMoment ::= SEQUENCE {
        minute                  Minute,
        millisecond             Millisecond
}

LaneChangeSpeed ::= INTEGER (0..500)

LeadingVehicle ::= INTEGER (0..4294967295)

FollowingVehicle ::= INTEGER (0..4294967295)

TargetLane ::= INTEGER (0..255)

TriggeringPointOfToC ::= INTEGER (0..10000)

AdviceLaneID ::= INTEGER (0..255)

AdvicePosition ::= INTEGER (0..10000)
```

```
DesiredBehaviour ::= CHOICE {
        targetGap               TargetGap,
        targetSpeed             TargetSpeed
}

TargetGap ::= INTEGER (0..255)

TargetSpeed ::= INTEGER (0..255)

TocAdviceReason ::= ENUMERATED {
        reason0 (0),
        reason1 (1),
        ...
}

PlaceOfStartTransition ::= INTEGER (0..10000)

TimeOfTriggerTransition ::= SEQUENCE {
        minute                  Minute,
        millisecond             Millisecond
}

PlaceOfEndTransition ::= INTEGER (0..10000)

END


ITS-Container {itu-t(0) identified-organization(4) etsi(0) itsDomain(5)
wg1(1) ts(102894) cdd(2) version(1)}

DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

ItsPduHeader ::= SEQUENCE {
        protocolVersion INTEGER {
        currentVersion(1)
        } (0..255),
        messageID INTEGER {
        denm(1),
        cam(2),
        poi(3),
        spat(4),
        map(5),
        ivi(6),
        ev-rsr(7),
        cpm(32),
        mcm(33)
        } (0..255),
        stationID       StationID
}

StationID ::= INTEGER (0..4294967295)
```

```
StationType ::= INTEGER {
               unknown(0),
               pedestrian(1),
               cyclist(2),
               moped(3),
               motorcycle(4),
               passengerCar(5),
               bus(6),
               lightTruck(7),
               heavyTruck(8),
               trailer(9),
               specialVehicles(10),
               tram(11),
               roadSideUnit(15)
} (0..255)

ReferencePosition ::= SEQUENCE {
       latitude                      Latitude,
       longitude                     Longitude,
       positionConfidenceEllipse     PosConfidenceEllipse,
       altitude                      Altitude
}

Latitude ::= INTEGER {
       oneMicrodegreeNorth(10),
       oneMicrodegreeSouth(-10),
       unavailable(900000001)
} (-900000000..900000001)

Longitude ::= INTEGER {
       oneMicrodegreeEast(10),
       oneMicrodegreeWest(-10),
       unavailable(1800000001)
} (-1800000000..1800000001)

Altitude ::= SEQUENCE {
       altitudeValue      AltitudeValue,
       altitudeConfidence AltitudeConfidence
}

AltitudeValue ::= INTEGER {
       referenceEllipsoidSurface(0),
       oneCentimeter(1),
       unavailable(800001)
} (-100000..800001)

AltitudeConfidence ::= ENUMERATED {
       alt-000-01(0), alt-000-02(1), alt-000-05(2), alt-000-10(3), alt-000-
20(4), alt-000-50(5), alt-001-00(6), alt-002-00(7), alt-005-00(8), alt-010-
00(9), alt-020-00(10), alt-050-00(11), alt-100-00(12), alt-200-00(13),
outOfRange(14), unavailable(15)
}

PosConfidenceEllipse ::= SEQUENCE {
       semiMajorConfidence           SemiAxisLength,
       semiMinorConfidence           SemiAxisLength,
       semiMajorOrientation          HeadingValue
}
```

```
SemiAxisLength ::= INTEGER {
        oneCentimeter(1),
        outOfRange(4094),
        unavailable(4095)
} (0..4095)

Heading ::= SEQUENCE {
        headingValue           HeadingValue,
        headingConfidence      HeadingConfidence
}

HeadingValue ::= INTEGER {
        wgs84North(0), wgs84East(900), wgs84South(1800), wgs84West(2700),
        unavailable(3601)
} (0..3601)

HeadingConfidence ::= INTEGER {
        equalOrWithinZeroPointOneDegree(1),
        equalOrWithinOneDegree(10),
        outOfRange(126),
        unavailable(127)
} (1..127)

Speed ::= SEQUENCE {
        speedValue             SpeedValue,
        speedConfidence        SpeedConfidence
}

SpeedValue ::= INTEGER {
        standstill(0),
        oneCentimeterPerSec(1),
        unavailable(16383)
} (0..16383)

SpeedConfidence ::= INTEGER {
        equalOrWithinOneCentimeterPerSec(1),
        equalOrWithinOneMeterPerSec(100),
        outOfRange(126),
        unavailable(127)
} (1..127)

LongitudinalAcceleration ::= SEQUENCE {
        longitudinalAccelerationValue         LongitudinalAccelerationValue,
        longitudinalAccelerationConfidence    AccelerationConfidence
}

LongitudinalAccelerationValue ::= INTEGER {
        pointOneMeterPerSecSquaredForward(1),
        pointOneMeterPerSecSquaredBackward(-1),
        unavailable(161)
} (-160..161)

LateralAcceleration ::= SEQUENCE {
        lateralAccelerationValue              LateralAccelerationValue,
        lateralAccelerationConfidence         AccelerationConfidence
}
```

```
LateralAccelerationValue ::= INTEGER {
        pointOneMeterPerSecSquaredToRight(-1),
        pointOneMeterPerSecSquaredToLeft(1),
        unavailable(161)
} (-160..161)

VerticalAcceleration ::= SEQUENCE {
        verticalAccelerationValue      VerticalAccelerationValue,
        verticalAccelerationConfidence AccelerationConfidence
}

VerticalAccelerationValue ::= INTEGER {
        pointOneMeterPerSecSquaredUp(1),
        pointOneMeterPerSecSquaredDown(-1),
        unavailable(161)
} (-160..161)

AccelerationConfidence ::= INTEGER {
        pointOneMeterPerSecSquared(1),
        outOfRange(101),
        unavailable(102)
} (0..102)

YawRate ::= SEQUENCE {
        yawRateValue    YawRateValue,
        yawRateConfidence       YawRateConfidence
}

YawRateValue ::= INTEGER {
        straight(0),
        degSec-000-01ToRight(-1),
        degSec-000-01ToLeft(1),
        unavailable(32767)
} (-32766..32767)

YawRateConfidence ::= ENUMERATED {
        degSec-000-01(0), degSec-000-05(1), degSec-000-10(2), degSec-001-
00(3),  degSec-005-00(4), degSec-010-00(5), degSec-100-00(6), outOfRange(7),
unavailable(8)
}

Curvature ::= SEQUENCE {
        curvatureValue CurvatureValue,
        curvatureConfidence    CurvatureConfidence
}

CurvatureValue ::= INTEGER {
        straight(0),
        reciprocalOf1MeterRadiusToRight(-30000),
        reciprocalOf1MeterRadiusToLeft(30000),
        unavailable(30001)
} (-30000..30001)

CurvatureConfidence ::= ENUMERATED {
                onePerMeter-0-00002(0), onePerMeter-0-0001(1), onePerMeter-0-
0005(2), onePerMeter-0-002(3), onePerMeter-0-01(4), onePerMeter-0-1(5),
outOfRange(6), unavailable(7)
}
```

```
CurvatureCalculationMode ::= ENUMERATED {
        yawRateUsed(0),
        yawRateNotUsed(1),
        unavailable(2),
        ...
}

DriveDirection ::= ENUMERATED {
        forward(0),
        backward(1),
        unavailable(2)
}

LanePosition ::= INTEGER {
        offTheRoad(-1),
        hardShoulder(0),
        outermostDrivingLane(1),
        secondLaneFromOutside(2)
} (-1..14)

SteeringWheelAngle ::= SEQUENCE {
        steeringWheelAngleValue                SteeringWheelAngleValue,
        steeringWheelAngleConfidence   SteeringWheelAngleConfidence
}

SteeringWheelAngleValue ::= INTEGER {
        straight(0),
        onePointFiveDegreesToRight(-1),
        onePointFiveDegreesToLeft(1),
        unavailable(512)
} (-511..512)

SteeringWheelAngleConfidence ::= INTEGER {
        equalOrWithinOnePointFiveDegree(1),
        outOfRange(126),
        unavailable(127)
} (1..127)

END
```

Annex B2: CAM ASN.1 specification

For the first iteration, TransAID is using the standard CAM ASN.1 definition [11].

Annex B3: DENM ASN.1 specification

For the first iteration, TransAID is using the standard DENM ASN.1 definition [13].

Annex B4: MAP ASN.1 specification

For the first iteration, TransAID is using the standard MAP ASN.1 definition.

Annex B5: CPM ASN.1 specification

For the first iteration, TransAID is using the CPM ASN.1 definition specified in MAVEN D5.1 [24].