D5.2 Simulated Baseline Cars

Project Number:	690705
Classification	
Deliverable No.:	5.2
Work Package(s):	WP5
Document Version:	1.0 2.0
Issue Date:	1.0: December, 27th 2017 2.0: November, 13th 2018 - Resubmitted
Document Timescale:	
Start of the Document:	
Final version due:	January, 31st 2018.
Compiled by:	Fabio Tango (CRF)
Authors:	Mohamed Cherif RAHAL (VED) Lynda Halit (VED) Maximilian Graf (ULM) Jürgen Pichen (ULM) Stefan Suck (OFF) Mark Eilers (HMT) Elisa Landini (REL) Fabio Tango (CRF) Alain Giralt (CAF) Paulin Pekezou Fouopi (DLR) Fei Yan (ULM) Martin Baumann (ULM)
Technical Approval:	Fabio Tango (CRF)
Issue Authorisation:	Andreas Lüdtke, (OFF)

DISTRIBUTION LIST		
Copy type[1]	Company and Location	Recipient
T	AutoMate Consortium	all AutoMate Partners

[1] Copy types: E=Email, C=Controlled copy (paper), D=electronic copy on Disk or other medium, T=Team site (Sharepoint)

Table of Contents

1 Executive Summary 5
2 Introduction 6
3 Baseline 8
3.1 ULM demonstrator (simulator) 8
3.1.1 Type of Support: H2A support in perception 10
3.1.2 Type of Support: A2H support in action 11
3.1.3 Relevance of the scenario 11
3.2 VED demonstrator (simulator) 12
3.1.4 Type of Support: H2A support in action 13
3.1.5 Type of Support: A2H support in perception and in action 14
3.3 REL demonstrator (simulator) 14
3.1.6 Type of Support: H2A support in perception 15
3.1.7 Relevance of the scenario 15
4 Implementation 17
5.1 ULM simulator 17
5.2 VED simulator 20
5.3 REL simulator 21
5 Conclusion and future Steps 25
References 26

List of figures

Figure 1: Schematic representation of the overall concept of the project...... 8
Figure 2 ULM driving simulator 20
Figure 3 ULM car Mock-up inside projection screen................................ 20
Figure 4: HMI with the "Automation deactivate" and "Automation activate" buttons in the Baseline car.. 22
Figure 5 VED driving simulator 23
Figure 6: REL driving simulator... 24
Figure 7: EVA scenario baseline script 25
Figure 8: Baseline HMI for Take Over Request 26

1 Executive Summary

This deliverable D5.2 "Simulated Baseline Cars" refers to Task 5.4, where the Baseline Car is developed. Thus, we describe here how the baseline cars are implemented in the driving simulators of ULM, VED and REL partners, with reference to the three scenarios of AutoMate project (that is, "Peter", "Martha" and "Eva").
In particular, we highlighted how the system should behave in the baseline and which is the benefits to have the TeamMate car implemented, for each scenario (and related use-cases). This gives also the rationale of our choice, when selecting a specific baseline and scenario.
Finally, we provided a technical description of the three driving simulators where the baseline have been implemented and evaluated.

2 Introduction

The top-level objective of AutoMate is to develop, evaluate and demonstrate the "TeamMate Car" concept as a major enabler of highly automated vehicles. This concept consists of considering the driver and the automation as members of one team that understand and support each other in pursuing cooperatively the goal of driving safely, efficiently and comfortably from A to B. As a consequence, in order to show how the enablers contribute to the implementation of this concept, it is important to briefly explain why the cooperation is needed, and how the human and the automation can support each other to create a safe, efficient and comfortable driving experience.

As shown in Figure 1, both the human and the automation have **limits** that can negatively affect the safety as well as the efficiency, the comfort, the trust and the acceptance of the autonomous driving.

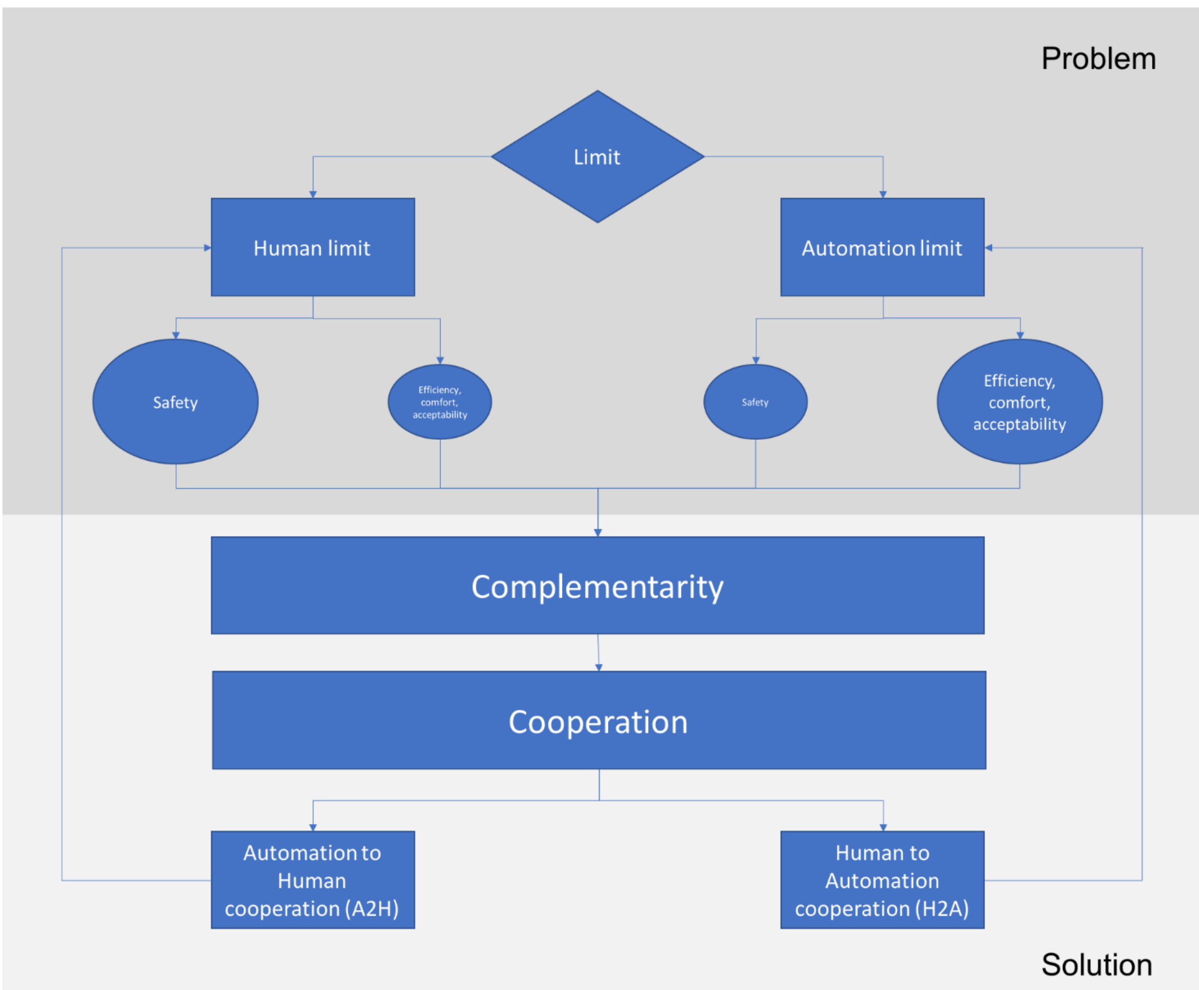

Figure 1: Schematic representation of the overall concept of the project

For the human, the limits are often related to his/her driving performance: they are likely to affect the safety, and cause accidents.
For the automation, the limits, mostly at perception and decision level, may affect the efficiency and the comfort of the trip, and then, in turn, the acceptance of the automation.
The AutoMate approach is based on the mutual complementarity between the driver and the automation: this support is achieved through the cooperation between the team members.
While the Automation to Human Cooperation (A2H) is used to complement the human limits, the Human to Automation Cooperation (H2A) is implemented to allow the driver to support the automation to overcome its limits.
The complementarity between the driver and the automation is the conceptual solution to compensate the reciprocal limitations, while the cooperation is how the complementarity is implemented. shows how both the A2H and the H2A cooperation can be implemented in perception (state A and B) and in action (state C and D).
This document is related to the activities carried out in task T5.4: "" in particular, the goal of this deliverable D5.2 "Simulated Baseline Cars" is to report and describe the prototype Baseline simulators, as the result of the activities carried out in T5.4. In details, it aims at developing Baseline Car (BLC) and integrate TeamMate (TM) technologies to develop the demonstrators of REL, VED and ULM, as driving simulators.
The scenarios and use cases, described here, are selected to demonstrate the relevance of the BLC solutions and, therefore, they are representative and consistent with the direction of cooperation implemented by that enabler, as well as the modality of support (i.e. either in action or perception).

3 Baseline

According to the refinement of the concept, new needs emerged to represent the cases in which the different types of cooperation is required and therefore this affects also the definition and the implementation of the baseline(s).

The scenarios in which the use cases are placed are the same as the first cycle; the use cases listed below describe different types of cooperation, i.e. the **two directions** of the cooperation (Automation to Human and Human to Automation) and the **two levels** of the cooperation (in **perception**, which implies giving a support but staying the in same state, and in **action**, which requires an active shift of the paradigm, i.e. a transition of control).

This approach, based on the theoretical framework described at the beginning of the project (cfr. D1.1), aims to implement the project's concept on the task distribution and the human-automation cooperation concepts.

3.1 ULM demonstrator (simulator)

The ULM simulator demonstrator is evaluated within the PETER scenario. The PETER scenario has been identified and selected since it is representative of a limit of the automation: overtaking on a rural road is in general a safety-critical manoeuvre and current automation will not perform this manoeuvre unless a very high confidence about the traffic situation ahead can be achieved. Therefore, in most if not all cases the automation is not able to safely overtake a tractor, because its sensors cannot acquire a complete view of the oncoming traffic situation (due to the tractor). As a consequence, the automated vehicle will stay behind the slow driving large vehicle, in our case the tractor, along the rural road.
In order to improve the efficiency of the automation behaviour, the automation can ask the driver for support (H2A, either in perception or in action).
H2A in perception aims at demonstrating how the human driver can support automation by taking over tasks at perception level and providing the information to the automation. In this use case, the demonstrator needs Peter's input to fill in missing information beyond its perceptual horizon, which is obstructed by a tractor. If this information is provided by the human driver, the overtaking manoeuvre can be initiated and carried out in a safe manner under the full control of the automation.

In this use case, the automation is in charge of the vehicle control, but it needs a support in perception from the driver to start a manoeuvre as essential information in its environment model is missing due to sensor limitations. As the automation can keep the system in a safe state even without this information (namely, following the tractor), it can still keep the responsibility for the vehicle control. That is, a request to the human driver to take over control is not necessary and will not be issued.

This use case is highly relevant because it investigates and exemplifies interaction strategies that reduce the number of situations where a disengagement of the automation is necessary. Such disengagement situations represent a highly critical condition for the interaction between the driver and the automation as the driver has to get back into the loop completely before being able to safely perform the driving task manually again. With the disengagement the driver loses much of the support of the automation. And finally, if such a situation as the PETER scenario, that is easy to handle by a human driver, can only be solved by the driver by taking back control completely from the automation, the driver will view this as a major restriction of the automation and possibly as an automation failure. Both interpretations can severely reduce drivers' trust into the automation and its acceptance. Reducing and avoiding such situations will therefore increase trust and acceptance of the automation.

The PETER scenario is also particularly relevant in demonstrating the safety gain of the driver with a TeamMate car, because overtaking on a two-lane two-way rural road is in general a safety-critical event and this criticality is even exaggerated in situations as demonstrated in the PETER scenario where the sight on the road is reduced due to a tractor in front. In such situations a driver without the support of a cooperative automation, as the TeamMate car possesses, being left alone might underestimate the risk of the current situation and might initiate an unsafe overtaking manoeuvre. The ULM simulator demonstrator shows how such behaviour can be avoided by a cooperative, transparent vehicle automation that provides, if available, information about the traffic situation ahead and about the current risk of the situation and possible manoeuvres.

Therefore, for the ULM simulator demonstrator, both the use case for the support of the driver to the automation (H2A in perception) and the use case for the support of the automation to the driver (A2H in action) have been selected to evaluate the added value of the TeamMate approach (i.e. the cooperation).

The selection of both H2A support and A2H support (as well as the corresponding different use cases), will be evaluated using a baseline that represents a fully automated vehicle that is necessarily optimized for safety and that does not allow the driver to change the automation's actions without fully taking back control of the driving task, that is, disengaging the automation.

The following text provides 2 simple stories (adapted from the PETER use cases) to intuitively describe the scenario for the evaluation of the ULM simulator demonstrator.

3.1.1 Type of Support: H2A support in perception

Here, we describe the H2A support for ULM demonstrator.

Starting scenario
Peter is driving in a narrow rural road in automated mode. As the car is approaching the tractor the car's environment perception becomes impaired as the tractor is blocking the sensors.

Baseline
As the vehicle is missing necessary information in its environment model to assess the upcoming road situation, e.g., whether there is oncoming traffic, it will choose the currently safest manoeuvre, namely following the tractor. This will be done as long as the automation's environment perception does not allow an absolutely confident assessment that the overtaking manoeuvre is safe or until Peter takes over control of the driving task, disengages the automation and overtakes manually.

TeamMate Car

Based on Peter's previous behaviour in similar situations, the TeamMate car, using its intention recognition mechanism, infers Peter's current intention. In case Peter has not overtaken in similar situations before, the TeamMate Car will stay behind the tractor and will not bother Peter by asking for support in overtaking. If the TeamMate Car infers that Peter normally would overtake in this situation, it recognizes a conflict between Peter's inferred intention and its own current capabilities that do not allow to fulfil Peter's probable goals. The TeamMate car asks Peter for support: Check the opposite lane as I can't see it and tell me whether I can initiate an overtaking manoeuvre or not. After Peter has checked the lane and is sure that there is no oncoming traffic, Peter communicates this information to the automation, and the automation initiates the overtaking manoeuvre being in full control of the driving task and continuously checking the road situation, updating its environment model with information becoming now available, assessing the risk of the current situation given the new information and acting accordingly.

3.1.2 Type of Support: A2H support in action

Here, we describe the A2H support for ULM demonstrator.

Starting scenario
Peter is driving in a narrow rural road in automated mode. As the car is approaching the tractor the car's environment perception becomes impaired as the tractor is blocking the sensors. Peter is in a hurry, so he decides to perform the overtaking manoeuvre manually.

Baseline
The baseline car would let Peter overtake because Peter is disengaging the automation and thereby requesting the full control and taking full responsibility based on his very limited environment perception. The system does not provide information about its environment perception nor will it explain to Peter why it might too risky to overtake and it will not interfere with him. If there is close oncoming traffic a safety-critical situation is likely to emerge.

TeamMate Car
The TeamMate car will communicate the relevant parts of its environment perception model to provide Peter a sufficient explanation of its behaviour. If the TeamMate car chooses to follow the tractor, for example, as it senses oncoming traffic or, based on its digital maps, knows about a close narrow curve ahead, it will communicate this to Peter. This communication will be more salient if the TeamMate car's intention recognition mechanism assumes, based on Peter's previous and current behaviour, that Peter has the intention to overtake. In order to avoid the pending safety critical situation, the TeamMate car warns Peter about the oncoming car, clearly depicting the situation to enhance Peter's understanding of the current traffic situation and explaining Peter why the TeamMate Car warns him, using its multimodal HMI.

3.1.3 Relevance of the scenario

The PETER scenario has been selected because it can represent the possible failure (mostly in terms of efficiency) of fully automated cars in situations, in which the sensors (e.g. cameras, radar, LIDAR) of the self-driving car reach the limits and the vehicle is not able to gain a full perception of the environment. For example, Peter approaches a tractor in front of him (automated mode). As the tractor blocks the detection of the sensors, the baseline car can only drive behind the tractor for a long time at low speed, which greatly influences the efficiency of executing the manoeuvre and the acceptance of the system in general as it leads to a rather uncomfortable situation for the driver.

In addition, the selected PETER scenario can also represent the critical events for automated vehicles. For example, Peter disengages the automation, as he feels very uncomfortable in the situation the automation catches him in and he is very unsatisfied with the automation's behaviour based on his limited understanding of the situation. Consequently, he initiates the overtaking manoeuvre without being aware of all aspects and the current high risk of the situation.

3.2 VED demonstrator (simulator)

The VED simulator demonstrator will be evaluated by considering the MARTHA scenario.
The MARTHA scenario has been identified since it is representative of a limit of the automation: in case of roadworks, the automation may not be able to detect the lanes to safely drive in Automated Mode.
As a consequence, the automated vehicle may unexpectedly handover the control to the driver (the so called "disengagement") and this situation could represent a safety critical condition for the driver (as already explained in the previous sections).
In order to improve the efficiency of the maneuver, and avoid the disengagement, the automation can ask for support to the driver (H2A in action).
H2A in action was selected in order to demonstrate how human can support the automation when the automation reaches its functional limits. The support in action implies that one of the team member needs direct intervention by the other for a safe driving.
While the H2A use cases selected so far (for EVA and PETER) describe a support in perception, and thus are linked to efficiency, trust and acceptance issues, the H2A in action is also particularly relevant for the safety of the driver, because without his/her intervention, the TeamMate car is not able to continue driving in Automated Mode and it has to perform either a disengagement or a safe maneuver to stop the vehicle.
The MARTHA scenario is also relevant for the safety because it considers a use case where Martha is distracted, and she needs the support of the automation to guarantee her safety.
Therefore, for the VED simulator demonstrator, both the use case for the support of the driver to the automation (H2A in action) and the use case for the support of the automation to the driver (A2H in perception and in action) have been selected to evaluate the added value of the TeamMate approach (i.e. the cooperation).

The selection of both H2A support and A2H support (as well as the corresponding different use cases), requires the definition of 2 different baselines for the evaluation:

- For the H2A use case, the evaluation is aimed at demonstrating the added value of the driver, thus the baseline is the driverless approach (i.e. the autonomous driving without any intervention of the driver)
- For the A2H use case, the evaluation is aimed at demonstrating the role of the automation to promptly and efficiently address safety-critical conditions, thus the baseline is the manual driving (i.e. when there is no support of the automation).

The following text provides 2 simple stories (adapted from the MARTHA use cases) to intuitively describe the scenario for the evaluation of the VED simulator demonstrator.

3.1.4 Type of Support: H2A support in action

Here, we describe the H2A support for VED demonstrator.

Starting scenario
The car is driving in an extra-urban road in Automated Mode.

Baseline
Through the sensors, the vehicle detects that the lane markings are no longer visible in about 150 meters. This implies a system boundary for the vehicle as its lateral control algorithms depend on detection of lanes. Hence, the vehicle decides that it cannot continue the trip safely and issues a take-over request (TOR) 6 seconds[2] before disengaging. A late TOR is likely to affect the safety and the acceptance of the system negatively.

TeamMate Car
Through the V2I communication, it detects that there are road works in 1 kilometer. Since the TeamMate car knows that it will not be able to deal with this situation in automated mode, it asks Martha for a support in action: in particular, it asks Martha to handle vehicle control through the road work zone. Martha is attentive, and she takes over vehicle control until the end of the roadworks, when the TeamMate car can shift back to Automated Mode.

[2] At 90 km/h, a distance of 150 meters corresponds to 6 seconds.

3.1.5 Type of Support: A2H support in perception and in action

Here, we describe the A2H support for VED demonstrator.

Starting scenario
Martha is driving in Manual Mode. She receives an incoming call.

Baseline
The baseline vehicle (full manual) is not able to detect her distraction. Martha's distraction may, thus, affect the safety of the trip.

TeamMate Car
The TeamMate car detects that she is distracted, so it informs her about the risk she is running. However, she does not care about the warning, and keeps talking animatedly on the phone. So, the TeamMate car informs her that it will take the control of the vehicle in a few seconds, and then it automatically shifts to Automated Mode.

3.3 REL demonstrator (simulator)

The selection of the H2A use case as the most relevant for the evaluation (to demonstrate the added value of the cooperation in the EVA scenario) also affects the definition of the baseline for the REL demonstrator. Since the demonstrator is aimed to show the value of the driver to support the automation, the baseline is represented by a condition where the driver has no role in the cooperation (i.e. the so called "driverless" approach): therefore, the baseline is the autonomous driving without any support of the driver.

The baseline has been defined by considering the elements that show the benefits of the TeamMate car against the baseline itself in the EVA scenario.

In particular, the benefits of the TeamMate car are:

- the vehicle could reduce the time that is needed to enter the roundabout (and, as a consequence, the frustration of the driver)
- The support in perception is able to increase the effectiveness of the trip
- the cooperation is able to improve the comfort and the acceptance

The following text provides a simple story (adapted from the EVA use cases) to intuitively describe the scenario for the evaluation of the REL demonstrator where the baseline will be used.

3.1.6 Type of Support: H2A support in perception

Here, we describe the H2A support for REL demonstrator.

Starting scenario
The car is driving in Automated Mode.

Baseline
When it approaches a roundabout, it detects high traffic flows. The car waits a lot before entering the roundabout, since (like Google car), it needs a relevant threshold of space to perform the manouver (see https://www.youtube.com/watch?v=Cnyq26N5tg0*). The car enters the roundabout after a lot of time, causing as a consequence relevant frustration to Eva.*

TeamMate Car
When it approaches a roundabout, it detects high traffic flows that can affect the efficiency (i.e. the TeamMate car evaluates that it may take some time to enter the roundabout in Automated Mode). To speed up the maneuver, the TeamMate car asks Eva a cooperation in perception, asking her to check the available space and to provide a trigger to start the maneuver. Eva checks the traffic and gives the confirmation to enter the roundabout. The TeamMate car understands the feedback and enters the roundabout in Automated Mode.

3.1.7 Relevance of the scenario

The EVA scenario has been selected because it represents an evident example of possible failure (mostly in terms of efficiency) of driver-less cars. In particular, findings show that in this driving situation the limit of the system is in the sensors, and in particular in the perception of the environment.
The behavior critical events for autonomous vehicles in roundabouts can occur in two phases, for different reasons:

- **Entrance:**
  The vehicle should be doing a look-ahead to anticipate a roundabout. Sensors (GPS or radar) only it is not yet possible to detect all elements that are relevant for autonomous driving (Rodrigues, 2017). So, for several systems, sensor data is combined with the detailed data of a

digital map. A problem that arises when a fully autonomous driving car uses map data is that the data must be 100% correct.

Moreover, there are many different kinds of roundabouts, so the currently developed techniques are still not able to correctly recognize every single roundabout (Perez Rastelli, 2015).

The sensors of the self-driving car are not always able to penetrate whatever is in the middle of the roundabout, and so it limits an ability to predict traffic patterns. The cameras of the self-driving car and the LIDAR and radar are only able to get a partial indication of what traffic is on the other side of the "island", and not able to gain a full sense of the other cars that are then coming ultimately toward the self- driving car via the concentric circles. It is still worthwhile to try and get that data in real-time and analyze it, but the AI of the self-driving car has to assume that the data will be noisy, obscured, and only provide at best partial information about what is taking place on the other side of the island.

- **Inside the roundabout**

Upon arriving at an entrance to the roundabout, the system needs to use its sensors to ascertain how many lanes are combined into this particular entrance. If there is only one lane, then the self-driving car has a simplified task since it only needs to focus on getting into the circular traffic that is flowing around the roundabout. If there are more than one lane, the car should be able to detect the lines on the road and select the most appropriate behavior, with a trade-off between safety and efficiency.

4 Implementation

In this section, the details of the implementation of the Baseline Cars in the driving simulators of ULM, VED and REL (respectively) are illustrated..

5.1 ULM simulator

The baseline for the ULM Demonstrator is a car defined as a highly automated vehicle SAE level 3, without any TeamMate features. This baseline will be used for evaluation against the TeamMate car features, which will be implemented in a simulator and a real vehicle. This allows the testing of various developments, e.g. HMI-versions, which can all be implemented in the simulator and only partly in the vehicle due to hardware restrictions. Therefore, in the following sections we divide into the simulator and vehicle implementations.

For the evaluation of the TeamMate car features, the baseline will be implemented in the ULM driving simulator. The driver will be able to interact with the system through a central touch panel. This GUI allows the user to choose between different actions via touch buttons on a very simple GUI in the central stack.
The baseline will be implemented in the ULM driving simulator with the SILAB driving simulation engine. The simulator is a mock-up that represents a real car (as shown in Figure 3) with a driver and a passenger seat. Additionally, there are several features in the driving simulator:

- steering wheel (force-feedback)
- pedals
- indicators
- central touch panel
- displayed rear mirrors (central, left, right)
- Smart-eye camera (static eye tracking system)

It also includes three high definition beamers that project the simulated environment onto a projection screen in front of the driver to create an immersive driving environment (as shown in Figure 4).

Figure 2 ULM driving simulator

Figure 3 ULM car Mock-up inside projection screen

It will be used to implement the baseline as well the relevant TeamMate car features for both situations (H2A and A2H).

SILAB is a highly customizable simulator engine that consists of several data processing units (DPUs). A DPU consists of input and output channels that

can be connected with other DPUs using a simple text-based description language.
A DPU can have one or multiple purposes. The complexity of a DPU reaches from simple Boolean logic to complex vehicle dynamics or rendering calculations. A DPU can be developed using C++, Matlab, Ruby, and Java. A DPU written in Java is called JPU.

In SILAB, everything is a DPU. For example, the steering wheel input values are processed using a DPU that grabs the CAN-Bus data of the steering wheel and converts the values into a numeric value on an output channel. This output channel is then connected to a DPU that handles the vehicle dynamics.
With this setup, it is possible to extend SILAB with automated driving features. SILAB itself provides basic assistance systems, like a DPU implementing an Adaptive Cruise Control (ACC). On top of this, a Java-based automation was developed and integrated into SILAB. The automation itself is a single JPU. The automation is divided into several components:

- An ACC wrapper that uses SILAB's ACC to handle longitudinal low-level control over the vehicle.
- A Lane Keeping Assist System that handles lateral low-level control over the vehicle.
- A road topology logic that analyses the lanes, for example if overtaking is allowed on a specific road segment.
- A behaviour logic that handles overtaking and lane change manoeuvres.
- A string-based command and control protocol to enable and disable several features of the automation or the automation itself

To control the automation itself, another DPU or JPU can be implemented that displays a graphical user interface and connects to the automation input channel to enable and disable the automation using the command and control protocol.

In order to allow the transition from automated to manual driving (that is part of a common interaction strategy for highly automated vehicles, and indeed part of the Baseline scenario) a Basic HMI (see Figure 6) has been integrated in the driving simulator. In the centre console, a 17"touch display (1024 x 1280px) is located allowing interaction with the automation. For the baseline car the interaction with the autonomous car is reduced to switch the system on and off. Therefore, the on/off button for the automation is displayed in the upper third of the screen. If the automation is turned off the button appears in a shiny blue and contains the in German written command "Activate Automation". By hitting the button, the automation will take over control and the button turn matt blue and the command switch to "Deactivate Automation". By pressing this button again, the automation will be turned off and the driver has to drive in manual mode again.

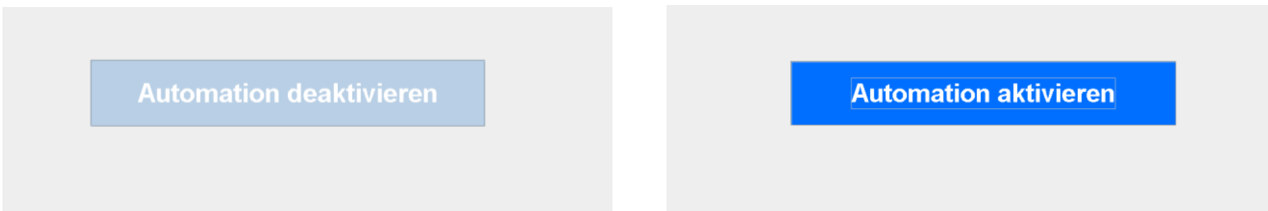

Figure 4: HMI with the "Automation deactivate" and "Automation activate" buttons in the Baseline car.

The interface was designed and implemented by using a JPU (written in JavaSwing 1.5). The graphical unit has been connected to the input channel of SILAB. This channel is connected to the Command and Control protocol of the automation software, which then executes the desired action.

5.2 VED simulator

VED has a static driving simulator composed of 4 x 32" screens displaying a total of 120° of horizontal field of view (as shown in Figure 6) while rear view is displayed by using three other screens. The driving simulator runs on Oktal's SCANeR^{TM} studio software.

Figure 5 VED driving simulator

It will be used to implement both baselines (automated and manual driving), because the software configuration for SCANeR™ studio software includes the following components, that allows the implementation of automated scenarios:

- 1 x SCANeR™studio Essential - Cluster configuration
- 1 x Terrain (3D environment edition + GeoData Import)
- 1 x Traffic and pedestrian model
- 1 x Driver module with basic human driver model and hardware interface with eye-tracker and physiological sensors
- 1 x Automated driving module
- 1 x CALLAS module to edit vehicle dynamics

1 x Europe v2 Driving environment representative of all road types in European countries

5.3 REL simulator

The baseline has been implemented in the driving simulator of REL based on a SCANeR™studio 1.7 driving simulation engine. The automated driving baseline has been implemented, in order to evaluate the H2A support. REL simulator consists in a 1-driver front passenger simulator with real controls and automotive parts:

- Force-feedback steering wheel
- Pedals
- Ergonomic seat

It also includes a projector to create an immersive driving environment.

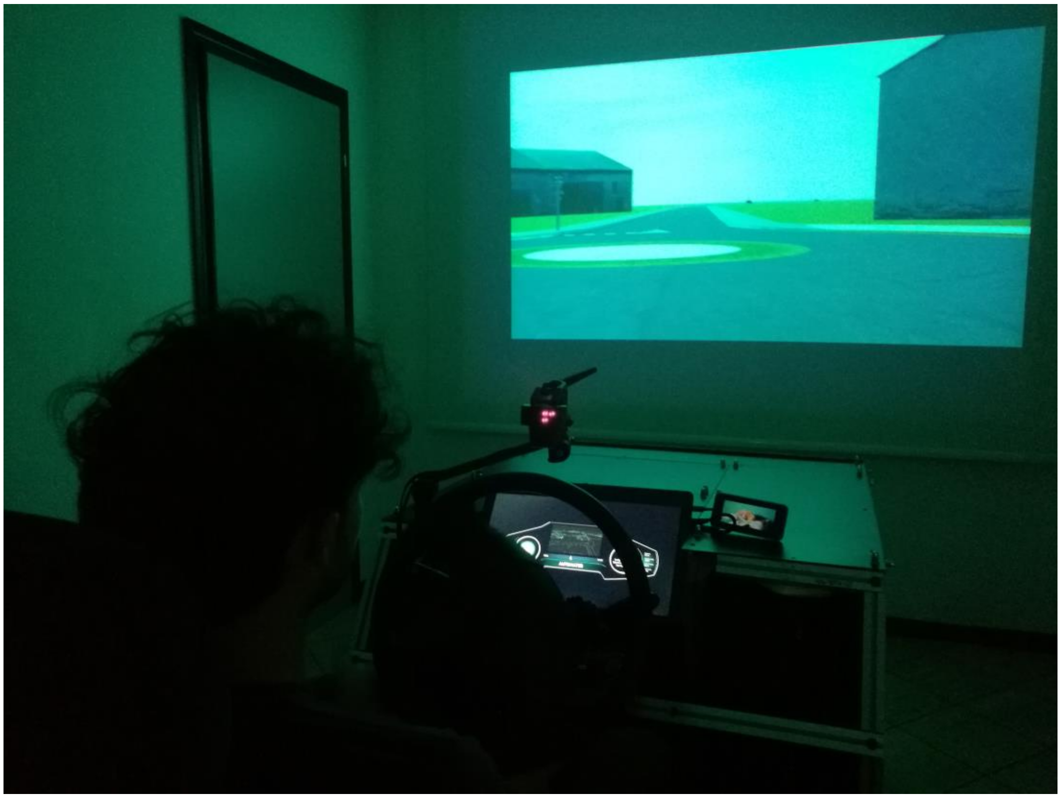

Figure 6: REL driving simulator

The software configuration for SCANeR™studio 1.7 includes the following components, that allows the implementation of automated scenarios:

- 1 x SCANeR™studio Essential - Cluster configuration
- 1 x ADD-ON Terrain (3D environment edition + GeoData Import)
- 1 x TRAFFIC & PEDESTRIAN MODEL
- 1 x ADD-ON AUTONOMOUS VEHICLE
- 1 x CALLAS Car RunTime (included in SCANeR™studio Essential)
- 1 x SENSORS

In the simulator, the baseline "Terrain", i.e. the environment in which both Baseline and TeamMate car will be evaluated.

Given this configuration, the simulator is able to perform every scenario autonomously. The simulated sensors, in fact, are able to reconstruct the environment and interpret correctly every situation.

The added value of using a driving simulator at this stage is the possibility to manipulate the situation in order to design the car behaviour according to the behaviour and the characteristics of real highly automated cars.
The scenario described in the previous chapter has been turned into a script to model the baseline car behaviour in the EVA scenario.
In order to model the car behaviour, the "Baseline script", consisting in a coded storyboard of the baseline scenario has been designed. It consists in a series of instructions given to a vehicle, in the form of events (e.g. triggers) and/or continuous events (allowing a change of state). Other simulated vehicles have been included in order to simulate a realistic driving environment. For each vehicle, including the ego-vehicle, a detailed itinerary has been created.

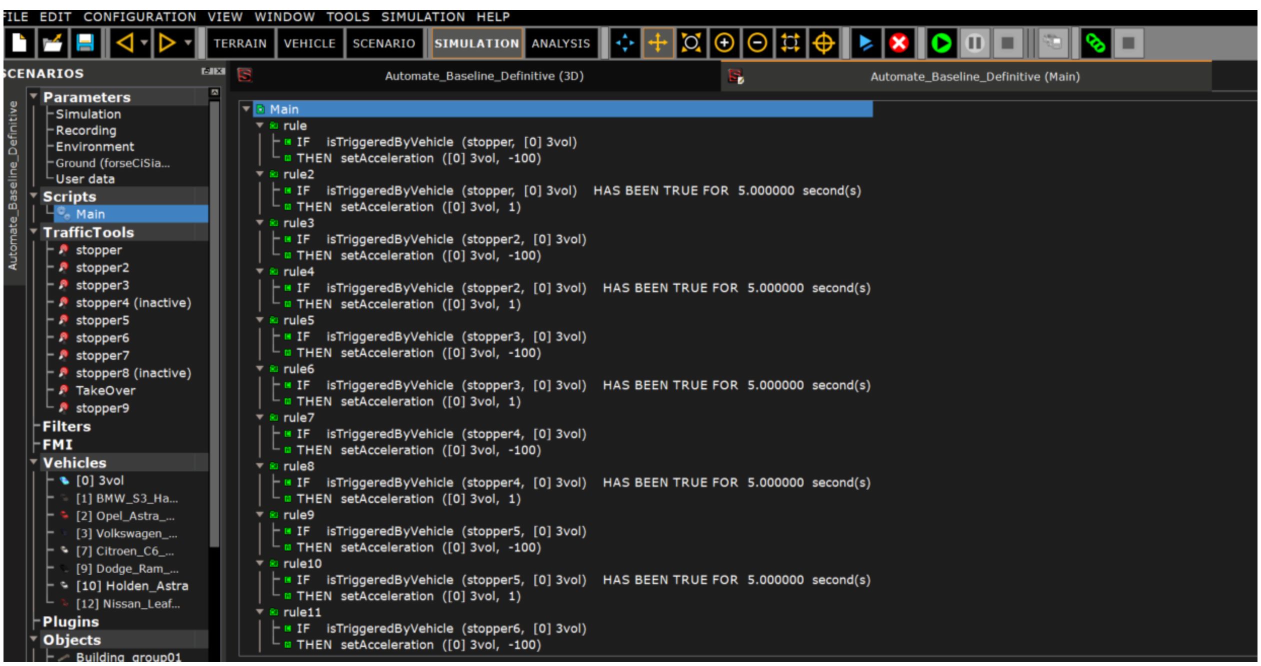

Figure 7: EVA scenario baseline script

Moreover, being the simulated car as default in fully autonomous mode, the simulator doesn't allow any information to the driver. It is provided with a default dashboard, showing only info related to the car behaviour (i.e. speed, gear, RPM): no information related to the control authority and no instructions are given to the driver.

In order to allow the transition from automated to manual driving (that is part of a common interaction strategy for highly automated vehicles, and indeed part of the Baseline scenario) a Basic HMI has been created to perform the Take Over Request. This HMI has been designed with a generic message (i.e. an icon that indicates to put the hands on the steering wheel and a text message), shown to the driver when a transition of control from automated to manual is needed.
Figure 7 shows the Take Over Request in the Basic HMI.

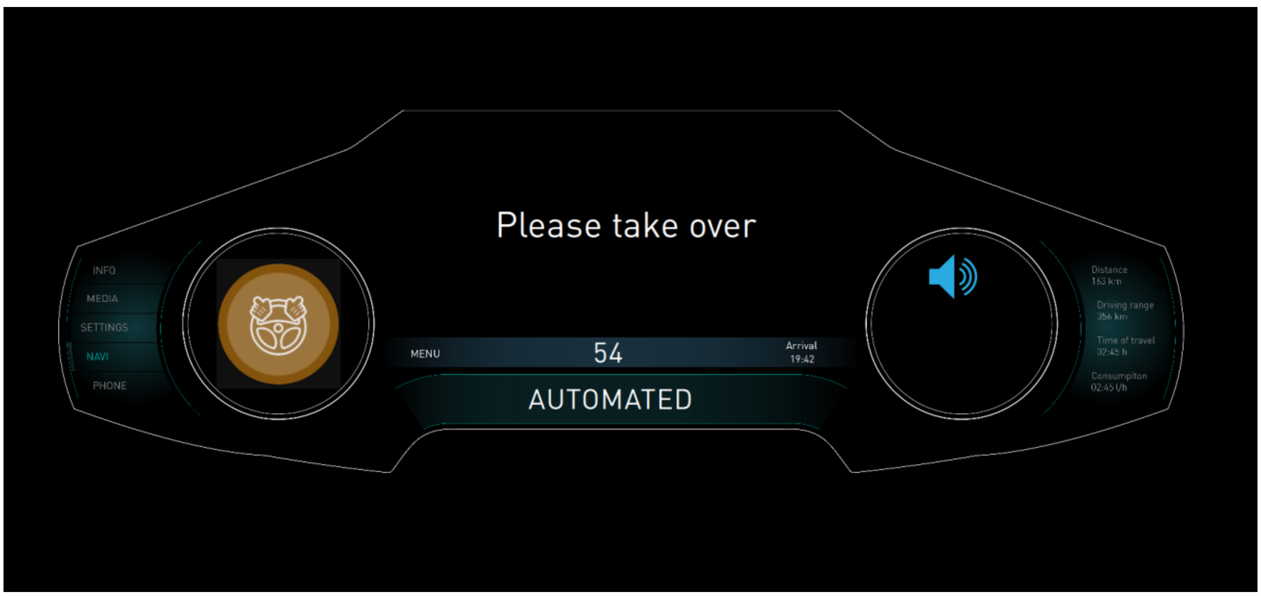

Figure 8: Baseline HMI for Take Over Request

This interface has been designed, implemented in Qt and connected to the driving simulator though UDP protocol. The display used to visualize the HMI is the same used to integrate the TeamMate HMI.

5 Conclusion and future Steps

This document illustrated the implementation of the simulated baseline cars, describing first the scenarios named “Peter”, “Martha” and “Eva” and, then, their implementation in the three driving simulators of ULM, VED and REL partners, respectively.
In particular, Section 3 presented the three scenarios, with a specific focus on the twofold type of interaction: human-to-automation (H2A) – when human-agent needs to support the machine-agent – and automation-to-human (A2H), when it is true the vice-versa (automation supports human driver). Based on that, the baseline is illustrated, where we show how the system behaves without the AutoMate system and which is the added value of the implementation of the TeamMate (TM) car concept.
Following this line, Section 4 details the implementation if the “different baselines”, for each scenarios and related use-cases, where also the driving simulators are described.

The next steps involve the implementation of the baselines on the real-cars of the project, as prepared by ULM, VED and CRF partners.
This will be described in the deliverable D5.5 “Real-vehicle Baseline Cars”, leaded by VED and due in month 28 (December 2018).

References

Rodrigues, M., et al., (2017), Adaptive behaviour selection for autonomous vehicle through naturalistic speed planning

Perez Rastelli, J., (2015), Fuzzy logic steering control of autonomous vehicles inside roundabouts