D8.6 - FINAL AUTOMOTIVE DEMONSTRATION

Deliverable ID	**D8.6**
Deliverable Title	**Final Automotive demonstration**
Work Package	**WP8 – Use Cases Implementation**
Dissemination Level	**PUBLIC**
Version	**1.1**
Date	**2019-12-30**
Status	**Final**
Lead Editor	**TTTECH, TTA**
Main Contributors	**Artiza Elosegui, Andreas Eckel (TTTECH, TTA), Melanie Schranz (LAKE)**

Published by the CPSwarm Consortium

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731946.

Document History

Version	Date	Author(s)	Description
0.1	2019-12-16	Artiza Elosegui, Andreas Eckel (TTTECH, TTA) Melanie Schranz (LAKE)	Initial text
1.0	2019-12-20	Artiza Elosegui, Andreas Eckel (TTTECH, TTA)	Final version ready
1.1	2019-12-30	Artiza Elosegui, Andreas Eckel (TTTECH, TTA)	Final refinement following review suggestions

Internal Review History

Review Date	Reviewer	Summary of Comments
2019-12-30	Farshid Tavakolizadeh (FRAUNHOFER)	Minor polishing, added few inline comments.
2019-12-30	Etienne Brosse (SOFTEAM)	Approved with minor comments

1 Executive summary

The present document is a deliverable of the CPSwarm project, funded by the European Commission's Directorate-General for Research and Innovation (DG RTD), under its Horizon 2020 Research and innovation program (H2020), reporting the results of the activities carried out in "Task 8.3 Automotive Use-Case" within WP8 – Use Cases Implementation. The main objective of the CPSwarm project is to develop a workbench that aims to fully design, develop, validate and deploy engineered swarm solutions. More specifically, the project revolves around three vision scenarios; Swarm Drones, Swarm Logistics Assistant and Automotive CPS.

WP8 aims at investigating application scenarios for the complete toolchain developed in CPSwarm. The work of this WP is be carried out in 4 tasks, one for each use case with a specific task dedicated to the use cases validation. Strongly driven by industrial needs, the work package is focused on three scenarios related to:

a) Swarms of drones and ground robots;

b) Swarm Logistics scenario;

c) Automotive use case.

This deliverable, namely "**D8.6 - Final Automotive Demonstration**", describes the work carried out in "Task 8.3 Automotive use case". The document is based on the predecessor document version "D8.5 - Initial Automotive Demonstration" and contains a few updates concerning the modifications and completion of work during the last work period.

This document provides final information on the conducted demonstrations and on how the developments carried out in this domain were evaluated and demonstrated. It covers the platooning use case both, in its implementation of the traffic simulation and in its implementation in the laboratory demonstrator.
The essential development seen from industrial point of view is the development of a safe and secure wireless communication channel capable of deterministic and reliable data transfer to be installed between the vehicles in the platoon.

Table of Contents

2 Introduction

Concerning the project lifecycle of CPSwarm, showed in Figure 1, the Experimental demonstration is a key part to get the final system delivery obligation of the project covered. This experimental demonstration is found within the work package number 8. It is divided into three main scenarios or use cases.

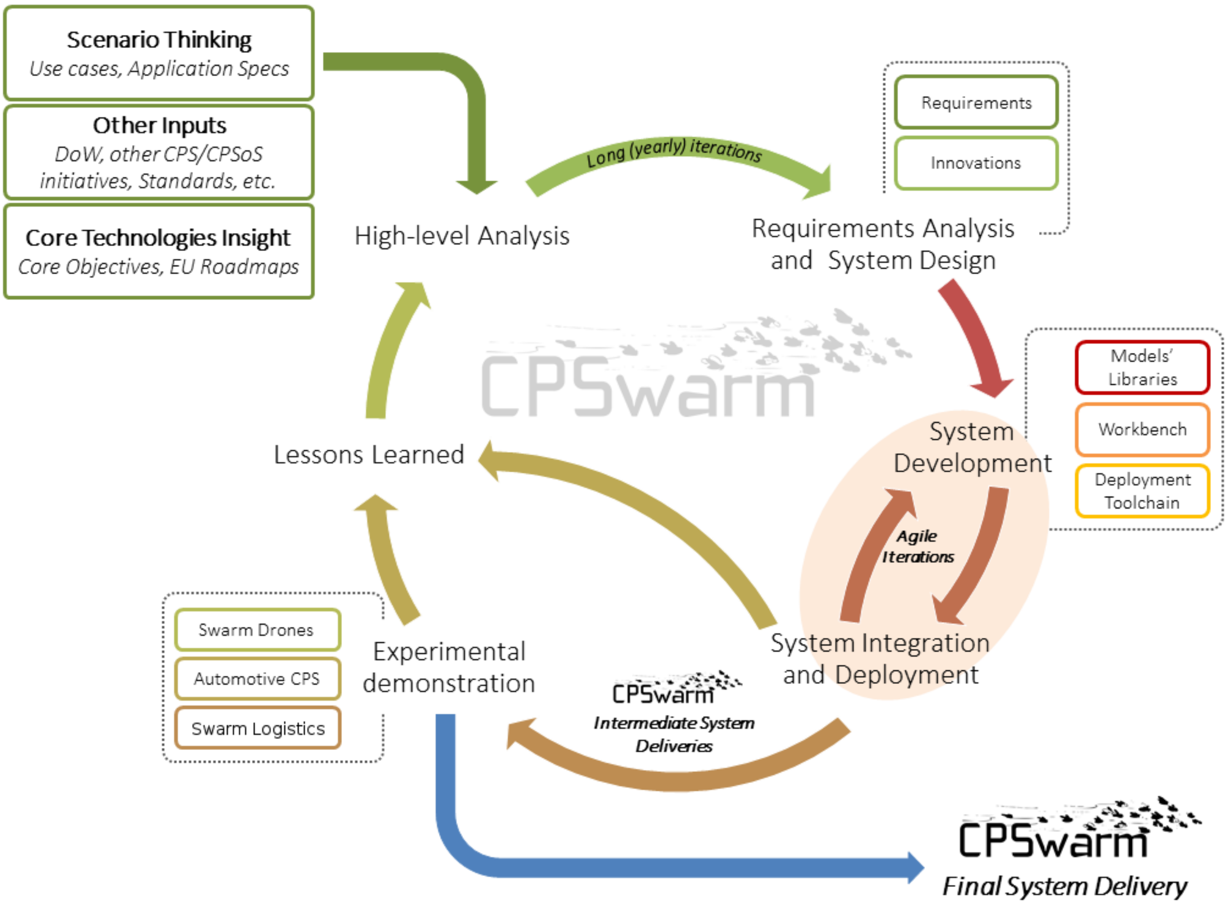


Figure 1: CPSwarm project lifecycle

The "**D8.6 – Final Automotive Demonstration**" document is a public deliverable focused on the results of *Task 8.3* Automotive use case *scenario* at the end of the project (M36). It provides the final descriptions on the automotive use case.

TTTech is the T8.5 leader and responsible for the delivery of D8.6.

2.1 Document organization

Generally, the aim of the **Final Vision** of this document series describes the work carried out in Task 8.5 and explains the target application scenario of the development and demonstrations conducted within the automotive use case scenario. D8.5 produced a first version while the final document D8.6 is enhanced by descriptions of the few final modifications carried out in the last work period. The work is based on the D2.2 which is dedicated to the scenario and the definition of the use case.

The document is organized as follows:

The Integrated Behavior (see §3.3): This section contains detailed descriptions about the platoon use case and how the platoon members interact.
The State Machine (see §3.4): This section contains the description of the state machine used for the simulation conducted in T8.5.
The Deterministic Wireless (WL) Driver (see §4): Contains the block diagram/description of the WL driver developed in this Use Case.
The Laboratory Level Demonstrator (see §5): The deterministic wireless driver is the core of the work conducted in the project with respect to the automotive use case. The Uses case as such shall only demonstrate the need of a wireless data connection and as such contains the description of the conducted demonstrations
The Simulation Framework (see §6): This section is dedicated to the description of the Simulation Framework established and the traffic simulation (in particular having in mind the traffic of vehicles on the road in contrast to "data traffic simulations and resulting measurements conducted on the wireless link) conducted within the frame of this part of the project work.

2.2 Related documents

ID	Title	Reference	Version	Date
[D2.2]	Final Vision Scenarios and Use Case Definition	D2.2	1.0	M16
[D8.5]	Initial Automotive Demonstration	D8.5	1.0	M24

3 Vision scenario

For this scenario, the vehicle platooning concept is addressed combined with swarm behavior and evolutionary algorithms.

3.1 Vehicle Platooning Concept

- The leading vehicle either has autonomous driving capability or is driven by human driver and prescribes the actions and decisions (i.e. navigation, decision on take-over maneuvers, sequencing maneuvers, lane change etc.) for the follow-up vehicles.
- The follow-up vehicles must have autonomous driving capability and environmental awareness. They follow the leading vehicle´s actions and receive vehicle date such as initiating brakes etc. via WLAN link (using the CPSwarm developed deterministic WLAN link).

Figure 2: Vehicles in platooning configuration.

Freight vehicle platooning holds great potential to make road transport safer, cleaner and more efficient in future. Platooning results in a lower fuel consumption, as the trucks drive closer together at a constant speed, with less braking and accelerating. Freight vehicle platooning has also the potential to reduce CO2 emissions. Likewise, connected driving can help improve safety, as braking is automatic with virtually zero reaction time compared to human initiated braking. Finally, platooning also optimizes transport by using roads more effectively, helping deliver goods faster and reducing traffic jams.

3.2 Description of the Vision scenario

As example, the following final vision scenario has been defined (although in practice the platooning configuration would most likely be only be used in long distance journeys and mainly in freight transport at high speed rather than in cities and mixed passenger car and freight vehicle traffic). This vision denotes a currently potentially not emphasized scenario just to demonstrate the potential up to "outer limits" of such application in future smart traffic. It is well understood that the major benefit is seen in smart freight traffic organizing efficient transport of goods.

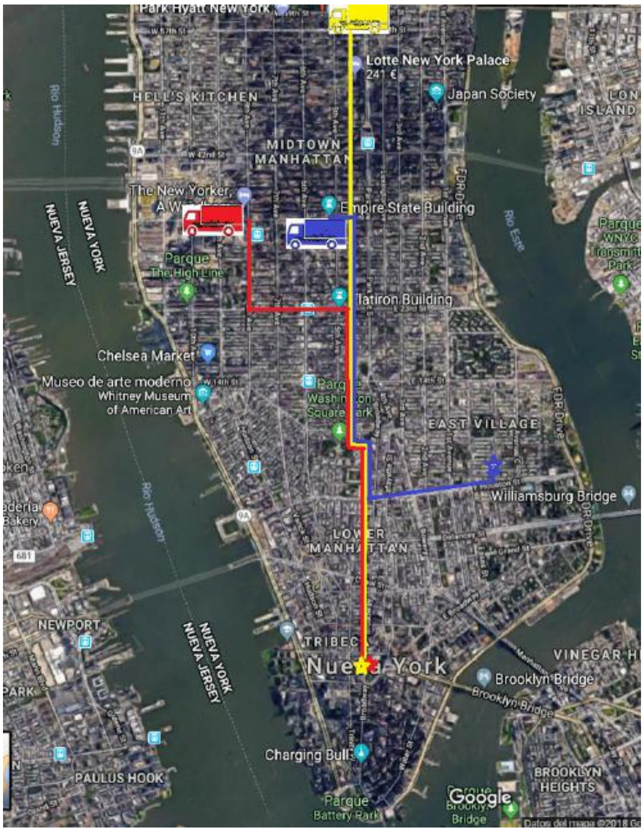


Figure 3: Example of 3 vehicles in a platoon

Referring to Figure 3, the yellow vehicle is an autonomously driving passenger transport vehicle that takes a group of tourists at The Hotel Plaza that want to visit the Tribute Museum. The red vehicle is an autonomously driving special goods transport vehicles and is at the Pennsylvania Train Station where it has picked up a new sculpture to bring it to the Tribute Museum. The blue vehicle is also an autonomously driving passenger transport vehicle that has picked up a group of travelers that have already visited the Empire State Building and want to go now to the Neyorican Poets Cafe located on the East Village, where they will have some rest.

Since a certain part of the route is common for the three vehicles, they decide to create a platoon. The blue vehicle joins the yellow vehicle as a follower whereas the yellow one leads the platoon. When they arrive at the Flatiron Building, the red vehicle joins them as a second follower vehicle. The three vehicles run together until Houston street where the platoon vehicles arrive at an intermediate destination for a “leave platoon maneuver” and thus where the blue vehicle leaves the platoon to go to its final destination. The yellow and the red vehicles keep the platoon until the Tribute Museum where they both reach their specific final destination.

3.3 Integrated Behaviour of Platoon Members

Although CPSwarm provides a lab demonstrator within the goals of the project only, the final vision for the vehicles is that they must have full autonomous driving capability thus they are capable of taking decisions on their own when, for example, an obstacle on the route does not allow them to continue with their route or stay within the platoon. In such a case they can change the lane (drive at left lane behavior) or brake to a full stop (emergency braking behavior).

When the vehicles run in a platoon configuration, the default assumption is that the leading vehicle uses its autonomous driving capability and prescribes the actions and decisions to be taken for the entire platoon thus also for the follow-up vehicles in the platoon (i.e. navigation, decision on take-over maneuvers, sequencing maneuvers, lane change, etc.). Although the follow-up vehicles have autonomous driving capability and environmental awareness, they follow the leading vehicle´s actions

Some vehicles might create a platoon while others will only follow the shortest path as fully autonomous vehicles. Vehicles that will create the platoon in the common route, will have to select their role (either leading vehicle or following vehicle) dynamically based on evolutionary algorithms.

On the other hand, when they create a platoon, due to the small distance among vehicles, some of the vehicles' sensors might become impaired (e.g. from camera); which means that they can only rely on the leading vehicle. The vehicles are connected via deterministic wireless communication (explained in §4) when they run in a platoon so that the leading vehicle can prescribe actions and decisions (i.e. braking/accelerating, lane change, navigation, decision on take-over maneuvers, sequencing maneuvers, lane change, etc.) for the follow-up vehicles. Relevant properties of such a distributed automotive system are modelled with the CPSwarm workbench.

When part of the route is common for two or more vehicles, they might create a platoon and while doing so even might be responding to the swarm intelligence. For example, they can commonly use routes that are less congested or avoid traffic jams that are ahead due to an accident or similar.

The mission describes the goal of the vehicle, i.e., where it is and where it should end. Of course, there might be numerous other constraints or parameters to be set, but this demonstrator's goal is to showcase why wireless communication is necessary not how to implement a platoon in detail. The optimization requested will be done on the route needed to execute the mission with the least cost. When two or more vehicles are traveling behind each other the cost of the mobility is reduced by 20% making platooning as preferred solution (i.e. in freight transport less drivers are needed (one driver per vehicle may be enough since he could also be resting during platoon drive and thus will not exceed maximum driving times), less fuel burned due to reduced drag, higher efficiency of road use, etc.). The goal of the optimization is to find out the best route for every vehicle. Therefore, the behaviors will be:

- The shortest path algorithm for each vehicle, from start position to intermediate destination (where a vehicle potentially will join or leave the platoon) up to potentially final destination per vehicle reached, responding to evolved or swarm algorithm provided by the CPSwarm workbench.
- Join/leave the platoon, responding to evolved or predictive algorithm provided by the CPSwarm workbench.

Some vehicles might create a platoon (see Figure 4) while others will only follow the shortest path as fully autonomous vehicles. The vehicles that will create the platoon in the common route, will select their role (either leading vehicle or following vehicle) dynamically based on evolutionary algorithms.
When they run in a platoon, the vehicles will be running with a given speed and with a given distance among them. The behaviors respond to situations of the real life, for example, when an accident occurs in the trajectory of the vehicles. In such a case, they are capable of taking decisions on their own and the behaviors will be:

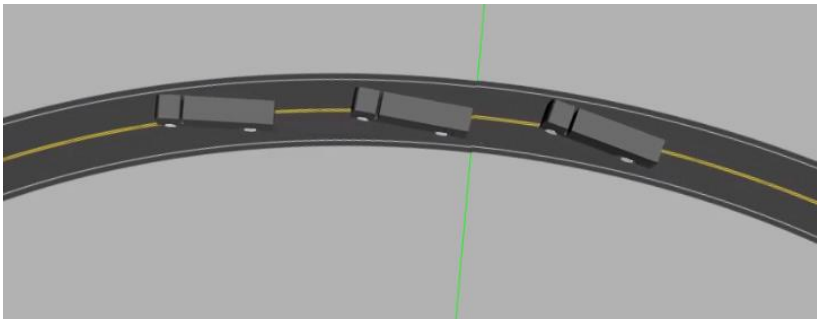


Figure 4: Platoon configuration

- Situation 1: Lane change.

  The road has multiple lanes and the leading vehicle changes the trajectory to the next lane on the left. The follow-up vehicles follow the leading vehicle always keeping the platoon configuration. There is no speed change. This is an event sent by the leading vehicle.

- Situation 2: Emergency breaking (see Figure 5).

  The road has only one lane and the leading vehicle reacts to the obstacle on the road by breaking until it completely stops. The follow-up vehicles break after the leading vehicle without collision. They must stop by keeping a minimum "safety" distance among them. The event is sent by the leading vehicles, instructing all vehicles to break.

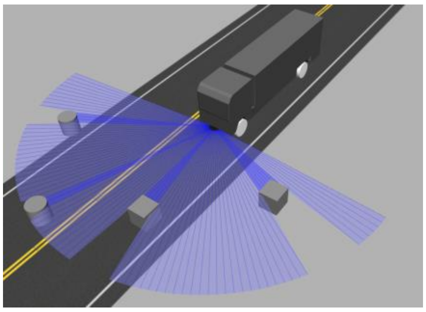


Figure 5: Obstacles detection for emergency breaking

3.4 State machine approach

Based on the vision scenario described above, the following state machine has been designed in collaboration with LAKE. The vision scenario makes use of two levels of states, level one denotes the highest level (level 1) where the major states denoting a platoon event are controlled such as Mission start/abort etc. (see Figure 6).

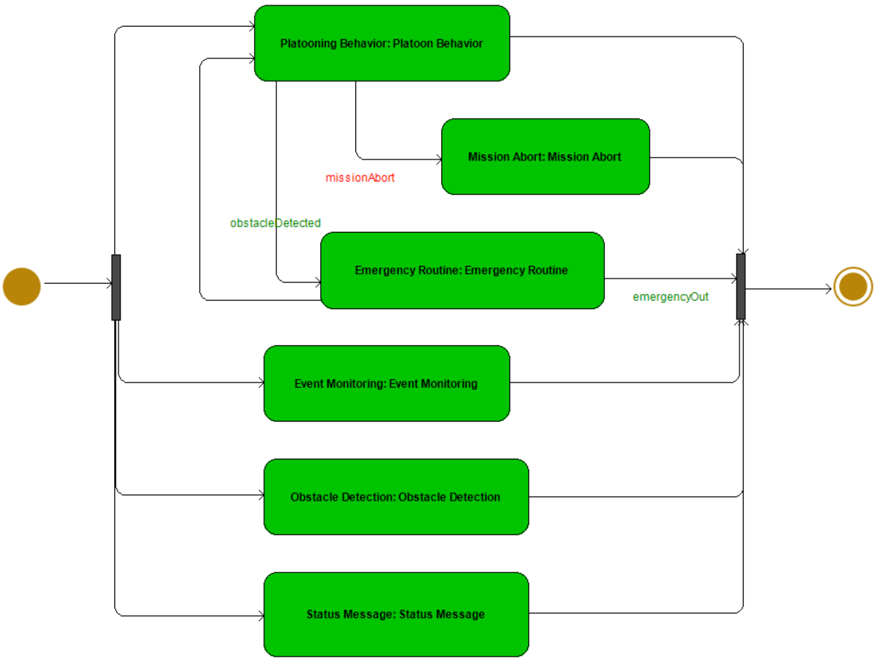


Figure 6: State Machine Demo, Model Scenario

Referring to Figure 6, it can be seen that the platooning function and control within this implementation is following a simple number of states defining the platooning behavior including mission abort and emergency states. In parallel the monitoring, obstacle detection and status message states are included in this simple example. This implementation does not emphasize to be a "ready to sell platooning" application the vision scenario shall only demonstrate the need of wireless communication among the vehicles of the platoon. Of course, a real product ready application will be much more detailed and complex.

For a complete list of level 1 states of this implementation is summarized in Figure 7.

High Level Events (level 1)

Identifier	Data	Sender	Scope
MissionStart		**MCT**	**Swarm Command**
MissionAbort		**MCT**	**Swarm Command**
MissionOver		**Swarm Member**	**Swarm**
LeadVehicle	**Vehicle ID, x/y coordinates**	**Swarm Member**	**Swarm**
RegularVehicle	**Vehicle ID, x/y coordinates**	**Swarm Member**	**Swarm**
ObstacleDetected	**Target ID**	**Swarm Member**	**Device**
BatteryLow	**Remaining Change**	**Swarm Member**	**Device**
Completed		**Swarm Member**	**Device**

MCT: Monitoring and Command Tool

Swarm Command	**red letters**
Swarm	**green letters**
Device	**black letters**

Figure 7: Events List (Level 1)

The level 1 state demonstration model is depicted in Figure 8. It shows the opportunity to integrate different swarm behavior options such as "Shortest Path Algorithm"," within the sequence of states in the platoon demonstration as well as parameter setting options such as "Select Role" and "Follow/Lead". The green boxes show the platoon related states while the yellow boxes represent the states for preparation of the platoon and free driving (no platoon).

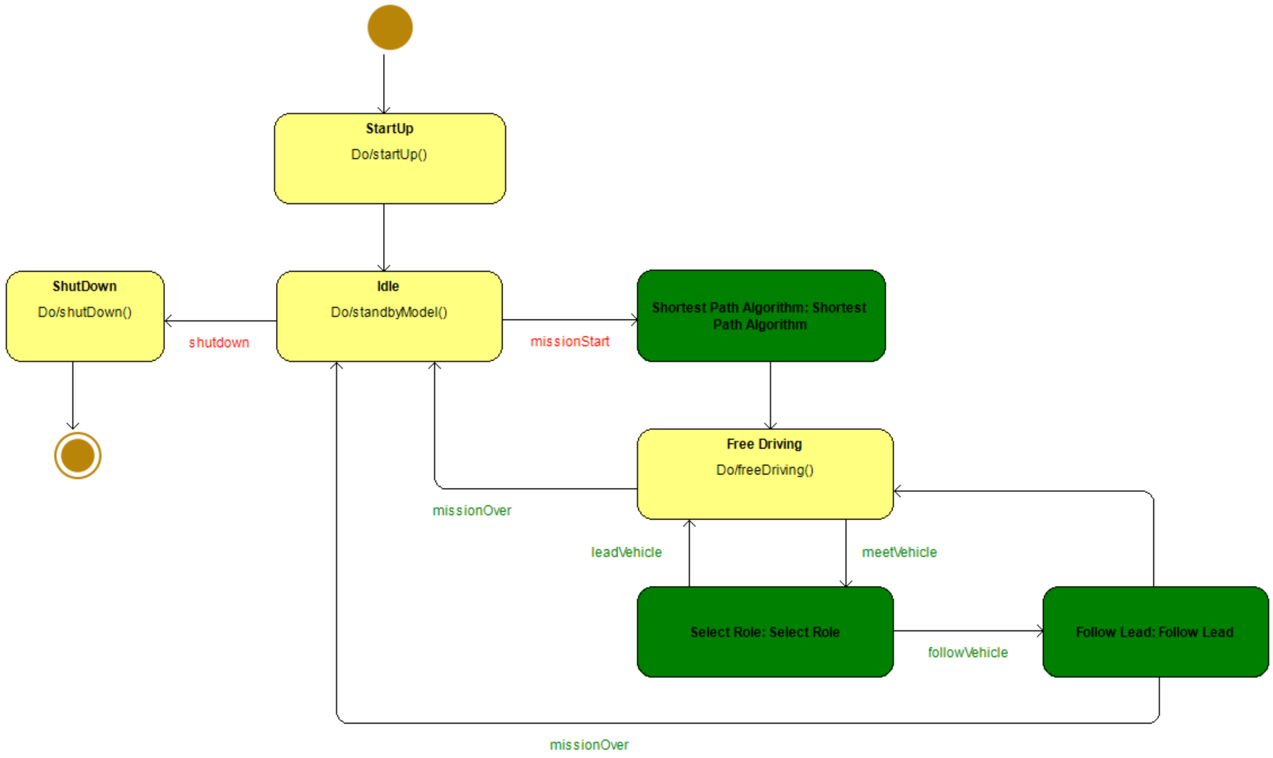


Figure 8: Level 1 Demo Model

The level 2 demonstration model shows the implementation of "shortest path algorithm" and "follow lead vehicles" (see Figure 9).

level 2:

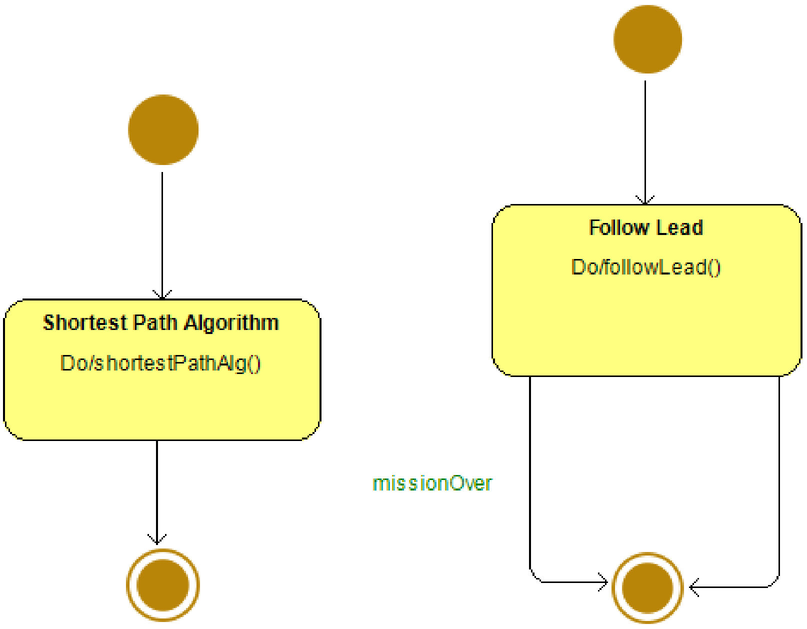


Figure 9: Level 2 Demo Model

Finally, the emergency routine model is depicted in Figure 10. As an example, it is packed into the lane change scenario, emphasizing – as a simplified approach -, to make a decision originated from the lead vehicle data transmitted to the platooning vehicles if a free lane is available or not. In case not, the emergency brake routine is initiated.

Emergency routine:

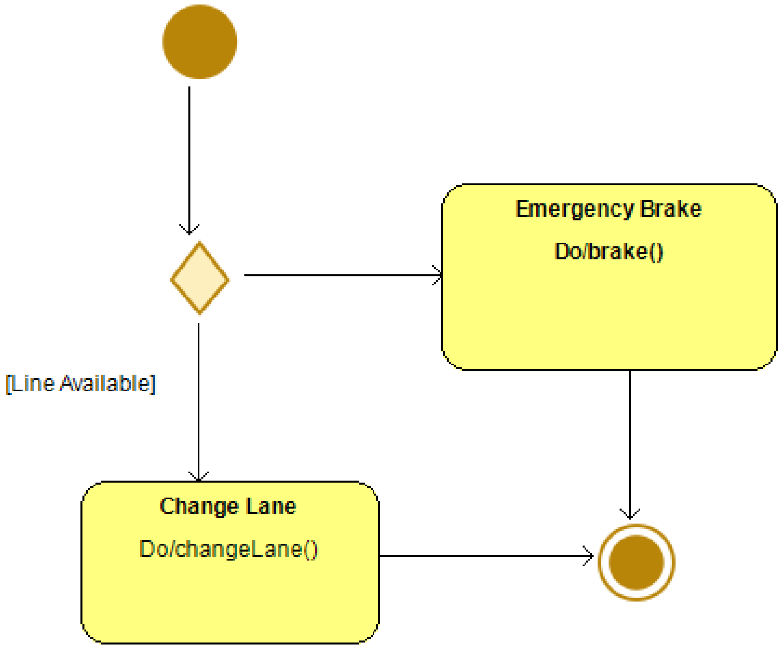


Figure 10: Emergency Routine Model

4 Deterministic wireless driver

Autonomous vehicles must communicate with each other without wired links while they are on route. The challenge therefore is to apply the know-how of the wired TTEthernet (SAE Standard AS6802) on a wireless environment. Due to the fact that a decision in the automotive industrial domain concerning the future wireless connection (i.e. 5G or V2X[1]) has not been made yet, the CPSwarm automotive use case uses WLAN as "easy to use and available" technology to try out to route data traffic using our TTEthernet protocol over this media. The gain expected from that was to investigate if at all a deterministic real-time and schedule-based data communication can be established over a wireless media keeping up guarantees for delay time while routing the data via WLAN (in future possibly other wireless) media. Deterministic data communication is seen as essential when it comes to safety-related applications such as autonomous driving or as emphasized here-in, platooning application.

TTEthernet is a scalable technology and allows development of critical system modules and applications according to fail-safe or fail-operational application requirements. As an example, a typical topology is depicted in Figure 11.

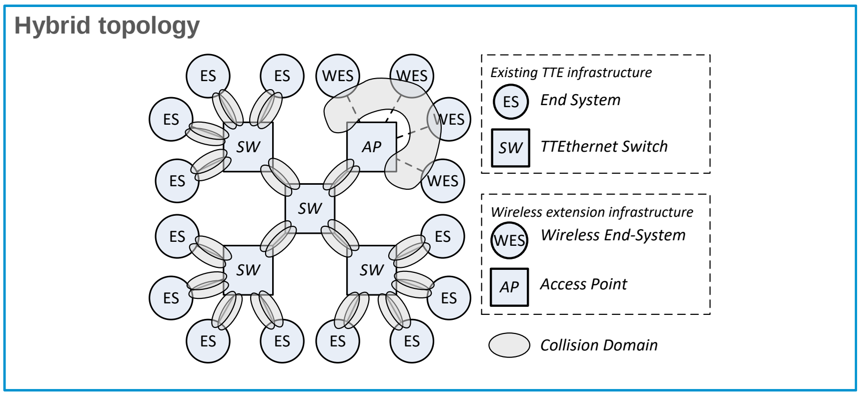


Figure 11: TTEthernet topology

The main difference between the wired and wireless links is seen in wireless links constituting a single collision domain when the stations are in range, whereas wired links are full-duplex connections.

To support integration of applications with different real-time and safety requirements in a single network, TTEthernet supports three different traffic classes:

- time-triggered (TT) traffic - is sent in a time-triggered way, i.e. each TTEthernet sender node has a transmit schedule, and each TTE-Switch has a receive and forward schedule. This traffic is sent over the network with constant communication latency and small and bounded jitter.
- rate-constrained (RC) traffic - is sent with a bounded latency and jitter ensuring lossless communication. Each TTEthernet sender node gets a reserved bandwidth for transmitting messages with the RC traffic. No clock synchronization is required for RC message exchange.
- best-effort (BE) traffic - traffic with no timing guarantees. BE traffic class compatible with the IEEE 802.3 standard Ethernet traffic.

CPSwarm endeavored to implement these traffic classes over a WLAN connection, understood as an example for wireless data communication. The initial assumption in CPSwarm was that porting our developments to other, potentially future technologies such as 5G or V2X, which might be selected as a media for automotive applications also in the safety-relevant area, confronts the developers with very similar problems as were found to overcome when using WLAN. However, already using 5G or V2X was not possible due to no mature enough and ready to use devices for both technologies were available at project start or significantly soon after the start to allow to use them.

---

[1] V2X: Vehicle to X communication, see https://en.wikipedia.org/wiki/Vehicle-to-everything

as technically viable and relatively straight forward.

4.1 Challenges of the automotive scenario

The major challenge was found in providing a guarantee for the delay of communication packets when sent over a wireless channel, which per se is not made for delay guarantees as provided by TTEthernet. Over the entire communication channel from the wired TTEthernet communication i.e. on the leading vehicle to the following vehicle and its internal wired TTEthernet network the guarantee for a delay time was finally implanted, even when found higher that without wireless communication route included.

4.1.1 Wireless communication

The communication from the leading vehicle to the follower vehicles, and among all other following vehicles in a platoon, must be mandatorily wireless since it is not possible to have a wire among vehicles when they are driving in a realistic scenario.

4.1.2 Real-Time communication

Real-time communication is compulsory to give response to the safety requirements, for example, when breaking. Network communication technology must use time scheduling to bring deterministic real-time communication fulfilling safety standards (i.e. ISO 26262 and AUTOSAR).

4.1.3 Handling low reliability communication

Real environmental environment like harsh weather conditions, obstacles or presence of other wireless signals may decrease the reliability of wireless transmissions and can compromise real-time communication requirements. Considering that the quality of the wireless channels varies with the time, frequency and location, it is possible to increase reliability by finding better times, frequencies and locations to transmit and/or by performing retransmissions, while still observing deadlines.

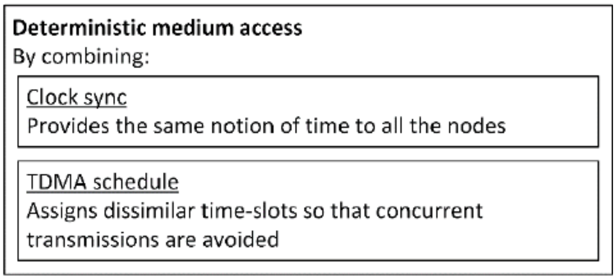
4.2 Wireless Driver integration

The wireless driver developed under CPSwarm meets all the requirements of the platooning scenario mentioned above. The communication driver is based on the IEEE 802.11 standard. As of today, wireless drivers compliant with IEEE 802.11 make use of the listen before transmitting Medium Access Control (MAC) principle in an attempt to avoid collisions between transmissions. However, this principle lacks real-time behavior, since transmissions can be postponed indefinitely waiting for the wireless medium to be cleared.

Our approach to guarantee “fairness” in the transmissions between vehicles consists of scheduling the points in time when every vehicle is capable to perform such transmissions, following a Time-Division Multiple Access (TDMA) approach. These instants are uniquely assigned to every vehicle so that collisions due to concurrent transmissions are avoided. For the schedule to be followed properly, a common time notion between the vehicles must exist. To solve this issue, a time synchronization protocol between the vehicles was deployed.

The deterministic WLAN implementation is based on the combination of two concepts (see Figure 12):

a) Clock Synchronization
b) TDMA Schedule



Deterministic medium access
By combining:

Clock sync
Provides the same notion of time to all the nodes

TDMA schedule
Assigns dissimilar time-slots so that concurrent transmissions are avoided

Figure 12: Media access with deterministic behaviour

Granting the access to the transmission medium is not enough for the transmissions to properly arrive at the destination vehicle, since reliability might be compromised. A common way of increasing reliability is performing retransmissions, so that the same data is sent several times. Retransmissions can be performed at different points in time or use different frequencies or physical paths. Our first attempt was to perform transmissions at different points in time while still considering deadlines and evaluate how good the solution performs for the vehicle environment.

This work was performed within the frame of CPSwarm T8.5. The API either includes normal sockets and addition configuration through standard driver calls, or alternatively, it uses custom API that wraps the sockets in the same way as the TTEthernet. Both options have two different traffic classes: best effort and time-triggered data communication.

Devices used for the driver testing (see Figure 13):

Figure 13: Test Devices

It is certainly understood that in a product-ready module for platooning the involved control electronics and algorithms would have to fulfil further safety and security requirements with respect to the networks and their architectures involved. Thus, this solution is understood as one brick (even a significant one seen as a “missing link”) that will enable modules used for autonomous or high automated driving to be extended to platooning applications. However, extending this development towards a prototype platooning module as a whole exceeds the means of the CPSwarm project.
During the conduct of the project and finalizing the development as well as evaluation work the following diagrams were recorded to demonstrate the results achieved (see Figure 14).

Figure 14: TTEthernet over WLAN performance

Figure 14 is used for showing the data packets transmitted at the intervals of transmission (upper diagram) and the related latency achieved over the TTEthernet over WLAN transmission channel. The diagrams show that, as expected, have a degradation compared to TTEthernet over wired connection but a limited variation allowing to emphasize use for safety-relevant application.
The bottom (third) diagram “Distance to Vehicle” is related to the platooning application and is described later in the document (see Section 5).

5 Laboratory level demonstrator

The automotive use case was implemented by a laboratory level demonstrator (TRL 3 to TRL 4, demonstration in breadboard lab environment) around autonomous driving vehicles connecting them via a platooning scenario implementation (kind of electronic drawbar) used to demonstrate relevance of the TTEthernet via WLAN development. It does not intend to be directly used as ready to use application product approach. Since the electronic platforms are mainly used for intra-vehicle computation, the aim of the demonstrator is to allow inter-vehicle communication to enable coordinated actions such as the ones described in the scenario. To do this, a wireless connection among the vehicle computing platforms must be established.

The demonstrator will make use of:

- Exemplary communication systems of two vehicles built in the lab demonstrator: One PC (potentially driving two displays, one for each vehicle) and two fog nodes (one per vehicle) with distributed Electronic Control Units (ECUs).
- Environmental awareness will be simulated for all of them using a simulator feeding live data from simulated sensors in potential integration of autonomous driving computers (not part of this project demonstrator)

The laboratory demonstrator consists of a representation of two trucks with identical equipment foreseen in a laboratory grade set-up. The idea is to have a central driving computer (The demonstrator uses one R-Car Device per vehicle) running in a trusted environment. The trusted environment is provided by means of a TTTech MFN 100 Fog Node virtualized computer. It also contains the WLAN interface and hosts the WLAN TTEthernet Driver developed within the frame of the CPSwarm project (See Figure 15 and Figure 16). The Fog Node also hosts a TTTech TTEthernet switch that can be used to connect further devices and nodes which then are also in the trusted environment providing protection against unauthorized access by means of virtualization and OPC-UA technology deployed.

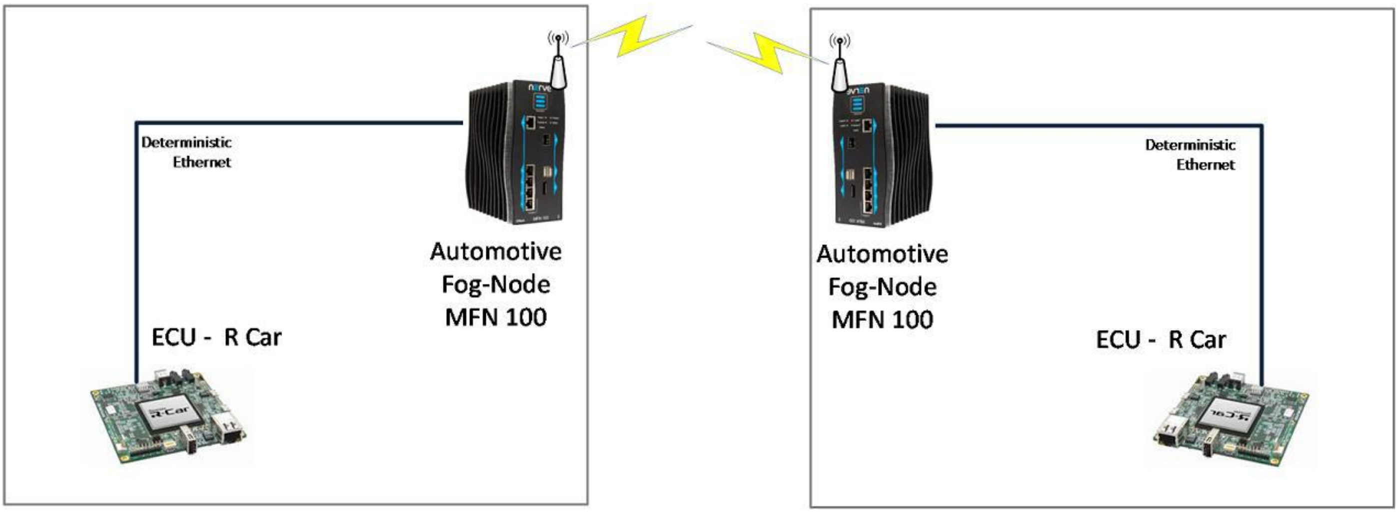


Figure 15: Block Diagram of the Automotive Demonstrator: TTTech's Fog node in the lab demonstrator

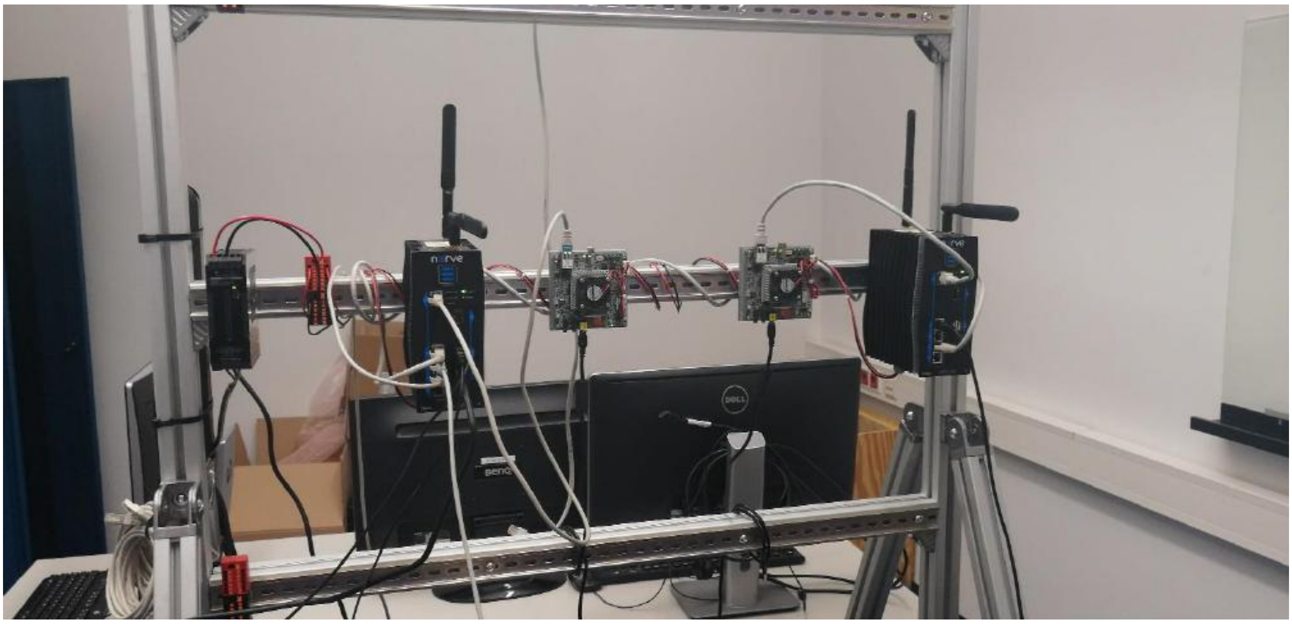


Figure 16: The implemented HW of the Automotive Scenario Demonstrator set-up

Furthermore, the fog node provides extra computational power and data processing within the vehicle and "offers" real-time processing power to other units in the vehicle, which might have rather restricted computing power. Data to be sent to the cloud can also be pre-processed so that the transmitted data is reduced by "data compression". The fog node also acts as a gateway between the Cloud and other end devices within the trusted environment established (ECU).

The Demonstrator consists of the following parts:

a) Demonstration of the wireless connection showcasing the safe/secure data communication path as a prototype verification set-up.
b) A "slow-motion" visualization of the data communicated as a proof of concept (i.e. the speed of the vehicles and the distance between them as well as acceleration parameter). It does not contain a full platooning application software since designing such application is not the expertise of the consortium partners. The demonstration puts focus on the data communication.
c) With reference to Figure 14 and the third diagram "Distance to Vehicle" provides a simulation about the benefit of having a direct deterministic, real-time data link between the vehicles based on a platoon braking scenario. In the first case, the breaking scenario is shown developing in slow motion. In the second case it can be recognized that the distance between the vehicles almost stays constant when the first truck initiates the brakes. This scenario uses the WLAN data connection where the information about brakes on in the leading truck is transmitted to the following trucks in the platoon, which then can initiate braking almost synchronously w/o significant delay. In the second scenario, the second truck only measures the distance by camera and thus loses the time between the brakes initiated at the first vehicle and the definite reaction of deacceleration having started (mechanical delay between first touch of brake pedal and braking reaction started is lost). This proves the effectiveness of transmitting the data rather than simply measuring the distance.

Hardware

The hardware of the automotive demonstrator consists of MFN 100 (fog node) and an R-CAR (ECU).

MFN 100 Edge Computing Device (see Figure 17)

Figure 17: TTTech Fog node

See the "The Role of Fog Computing in the Future of the Automobile" in the annex.

R-CAR ECU (see Figure 18)

Figure 18: R-CAR ECU

The R-CAR ECU is compliant with the ISO 26262[2] (ASIL-B) functionality safety standard for automotive and has enhanced security functions and improved robustness. The R-Car ECU can be applied to in-vehicle as driving safety support system.

5.1 Architecture

Applications deployed on the automotive demonstrator are typically distributed. Such part of the software can be deployed on the Fog-Node and part on the ECUs. For example, pre-processing of sensor data, which requires fast real-time response but not much computing power can be deployed on the ECUs and advanced object recognition algorithms on the Fog-node.

---

[2] ISO 26262 standard on functional safety, https://en.wikipedia.org/wiki/ISO_26262

The proposed architecture is depicted in Figure 19:

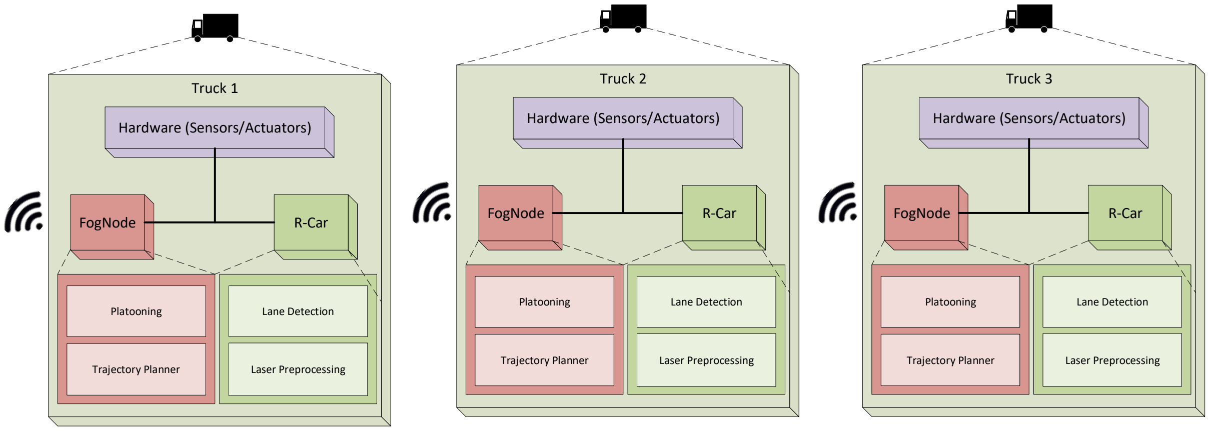


Figure 19: Architecture of the automotive use case

Each of the vehicles is considered as a black box and is responsible for each own sensors and actuators. Only the Fog Node is visible from outside.

The vehicles, by means of the Fog Nodes, can communicate with each other wirelessly. They can exchange information such as the speed of the vehicle, exceptional situations (e.g., emergency breaking), the fuel level or any other information crucial to the mission. However, they do not expose internal devices to the cloud (Trusted Environment).

The list of components and what will run where is shown below (see Figure 20):

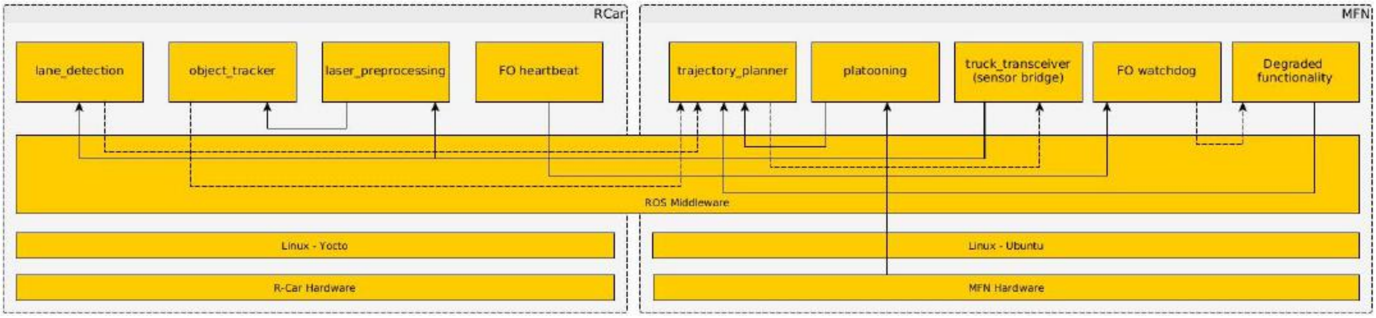


Figure 20: Components List

5.2 ROS node

The ROS node can (if required) either be integrated in the Fog Nodes or the R-Car node. However, this will be required only in a set-up that is closer to a product approach and this is not part of this set-up. Within the CPSwarm project an architectural concept is proposed, which would have to be discussed with a potential customer at implementation time (the approach depends on the application programming and on the safety architecture of the overall system i.e. the platooning application).

5.3 Failover (FO)

A failover mechanism may be installed in case the R-Car ECU fails. In such case the MFN 100 Fog Node can replace a potentially integrated ROS node or take over functionality hosted on the R-Car ECU.

The FO heartbeat is a component sending a heartbeat to a watchdog. If the heartbeat didn't arrive (because the R-Car ECU failed) the watchdog will boot up some graceful degraded software components in order to replace the applications initially running on the R-Car ECU.

6 Simulation framework

6.1 Network simulation

A cost effort constrained way to test whether the communication between vehicles perform in realistic environments with heavy vehicle traffic and wireless channel conditions as found in roads or city streets is to use a simulator. OMNeT++[3] discrete event simulator, in combination with the INET framework, allows for data communications network simulation. TTTech investigated the integration of a TDMA layer on top of the standard IEEE 802.11 modules in INET that allows to simulate a network in which transmissions are scheduled. To enable realistic vehicular traffic scenarios, SUMO[4] (Simulation of Urban Mobility) is used in combination with OMNeT++ inside the Veins framework. With Veins, the static TDMA nodes can be placed in vehicles, simulating the interactions that arise when the distance between vehicles varies, other vehicles appear on stage or buildings and other obstacles influence the wireless signal. The main result coming from the simulator is the reliability of the transmissions (e.g., percentage of lost data packets) and delays (e.g., time it takes from the moment the message is sent to the MAC protocol until it is received at the destination). The results can be used to, e.g., introduce modifications in the transmission schedule that can improve reliability and reduce delays for the particular simulated scenario.

The demonstration aims at showing the difference between a plain wireless network and a time-triggered network concerning the packet loss due to interference and collisions (see also Figure 14). Assuming a swarm of swarm members called "agents" that need to communicate with each other or with a central station through wireless, one needs to consider the packet loss due to collisions. As with any CSMA/CA network there is going to be collisions due to end systems starting to transmit at the same time. In a wireless network this goes a step further since there might be interferences from external factors, resulting in possible packet loses. Luckily mechanisms handle retransmission of these packets; however, this introduces delays in the network. This problem scales up with many different factors, as the agents start to move away from each other. To communicate they will have to relay their message via another agent resulting in a multi-hop network, for each hop there is a chance that a collision might occur. Sending big amount of data image also scales as the problem since if there is an error at the transmission the whole packet will have to be transmitted again resulting in delay. Finally, the most common factors are the number of the agents (since the more agents you have the more collision you will get) and amount of data (the more data you transmit the more collision you get again).

6.2 Demonstration explanation

In the first demonstration, a network with six agents named (A to F) is simulated (see Figure 21). The northern agents (A, B) communicate with the southern agents (E, F). Due to the distance the last two agents act as relays between them. The demonstrator was built to present the problem when working with wireless thus of course parameters were set to maximize the problem. The agents send medium size data to each other (imagine a list of detected objects, or some geographical data) in very short intervals (to save costs from implementation they don't check if the same data has been sent again so they just send everything again). As the demonstration is run, one will notice that the southern and northern groups tend to send messages at the same time. This is not a problem since they are so far away that they don't interfere with each other. However, the same does not go for the relays at the middle, where both are affected by the transmission of the two groups resulting in the agent not dropping almost all of the packets.

---

[3] See https://omnetpp.org/

[4] See https://www.dlr.de/ts/en/desktopdefault.aspx/tabid-9883/16931_read-41000/

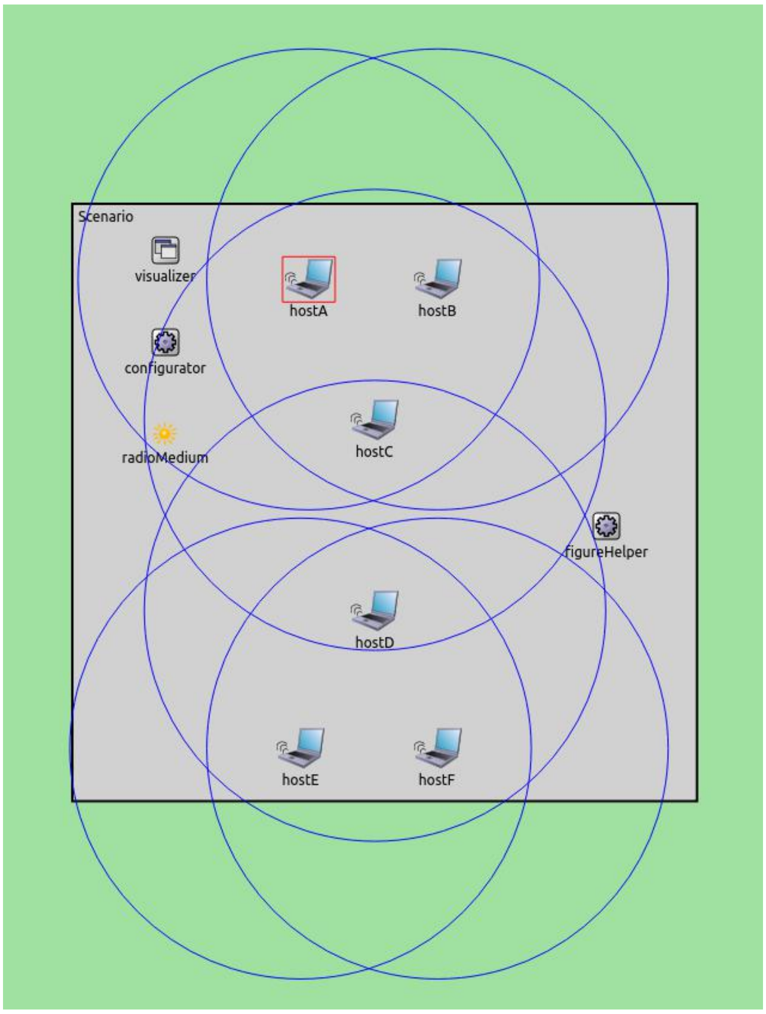


Figure 21: Demonstration with six Agents Scenario

As already mentioned, this is an extreme example were the traffic was so heavy resulting in this heavy packet loss. There are many ways to overcome these problems such as having a single drone as a relay (limiting the maximum distance), sending less amount of data (compression), not sending unneeded data (pub/sub protocol or specific implementations). However, one problem that is still not solved even with these methods is guaranteeing real time communication (having an upper bound of when the message will arrive). Why is the real time communication so important? An agent is patrolling at the edges of the wireless signal, suddenly the swarm decides to move the opposite direction and informs everyone due to interference that one of the agents did not receive the update. So, it is kept patrolling while the swarm moved away from the range (the swarm can still move back to its original location to reconnect with that agent). Let's take a look at an automotive scenario now. There is a platoon of ten trucks driving behind each other fairly close to each other to reduce drag and fuel costs, after a turn there is an accident and an emergency breaking is required. Due to the close following distance the front facing sensors would not detect the braking of the object fast enough to stop in time, it needs to be done via communication. In a platooning, the first one to break is actually the last one, since if the first one breaks each vehicle in the platoon will be progressively stopping closer to its front truck, possibly resulting in a crash. In this case, the message would have to be relayed through the whole chain until it reaches the end and then back again to start the braking. Imagine the message being transmitted successfully instead of the standard 2-4ms (per truck) to 100ms due to collisions.

6.3 Traffic simulation

During the first period of the project, Gazebo was the simulator used for the scenario simulation. Gazebo is a robot simulation tool requires the implementation of (1) completely functional autonomous driving system (lane detection, trajectory planning, object detection, collision avoidance) (2) control of the vehicle actuators (steering, throttle). However, to add missions, relevant improvements should be done to the current scenario:

- Lane detection for multiple lanes and turns (increased the complexity of lane detection);
- The trajectory planner should be able to distinguish and pick the correct lanes to follow;
- More refined control of the vehicle (accurate maneuvers);
- Route planning.

However, after several months, the decision to use SUMO (Simulation of Urban Mobility) was taken. SUMO is an open source road traffic simulation package designed to handle large road networks and is licensed under the GPL. Figure 22 shows the resulting traffic simulation.

Figure 22: Traffic Simulation using SUMO

SUMO brings the following benefits:

- Being a road simulator, a complete autonomous driving system or actuator control is not required;
- Building road networks is simple since they can be imported from OpenStreetMaps[5];
- Focus will be shifted to the platooning functions rather than the controlling the vehicle;
- By default, vehicles are given routes to follow.

Features of SUMO:

- Simple control of individual vehicles (speed, route, change lane, break);
- Large number of vehicles and huge maps can be supported;
- Sumo can be used in combination with other tools (VEINS, OMNeT++, Unity 3D[6]);
- Fuel consumption calculation.

Additionally, SUMO is compatible with other applications and simulators that can complement the traffic simulation:

- OMNeT++ is a discrete event simulator.

---

[5] See https://en.wikipedia.org/wiki/OpenStreetMap

[6] See https://de.wikipedia.org/wiki/Unity_(Spiel-Engine)

- The INET framework[7] is a model suite for wired, wireless and mobile networks.
- Currently used by TTTech Labs employees in this research (deterministic wireless driver).
- VEINS is an extension of OMNeT++ and is used to bridge SUMO and OMNeT++.
- SUMO handles the controlling of the vehicles.
- OMNeT++ handles the communication between the vehicles.
- Plexe[8] is an extension to VEINS and SUMO which adds platooning behavior to the simulation.
- Permit[9] is an extension to Plexe allows to simulate platooning maneuvers.

Example (see Figure 23): Platooning application on a highway

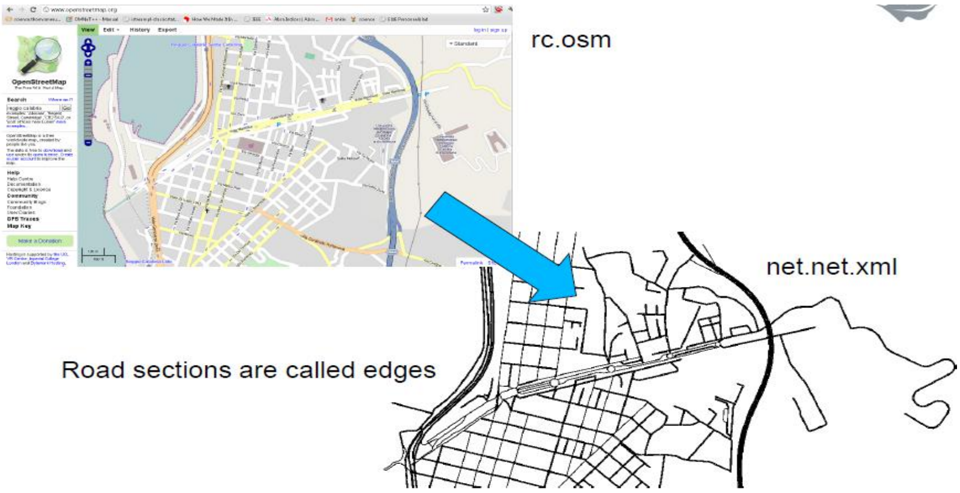


Figure 23: Platooning application on a highway

- VEINS framework: Integrates SUMO & OMNeT++.
- Simulation results → input for network design.
    - Mixed criticallity: safety relevant (e.g., schedule design), high volume, etc.
- *Integration in CPSwarm Workbench*

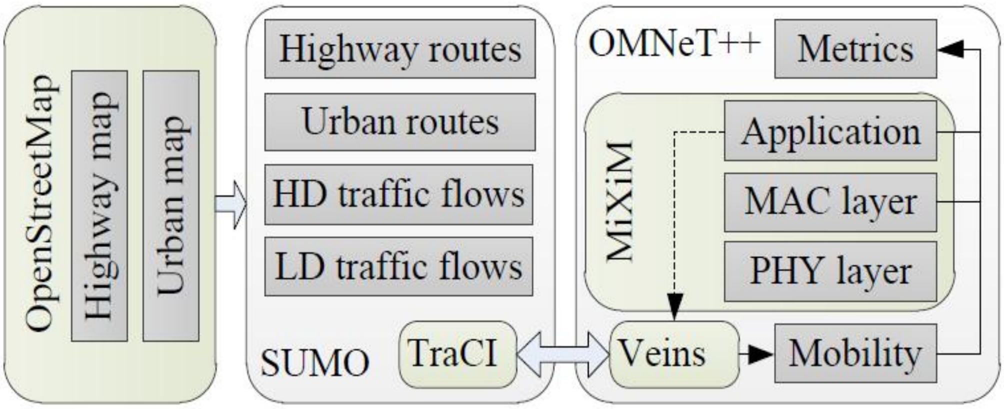


Figure 24: Platooning application on a highway - Model

The system simulation architecture is the following (See Figure 24):

---

[7] See https://inet.omnetpp.org/
[8] See http://plexe.car2x.org/
[9] See https://omnetpp.org/download-items/Plexe.html

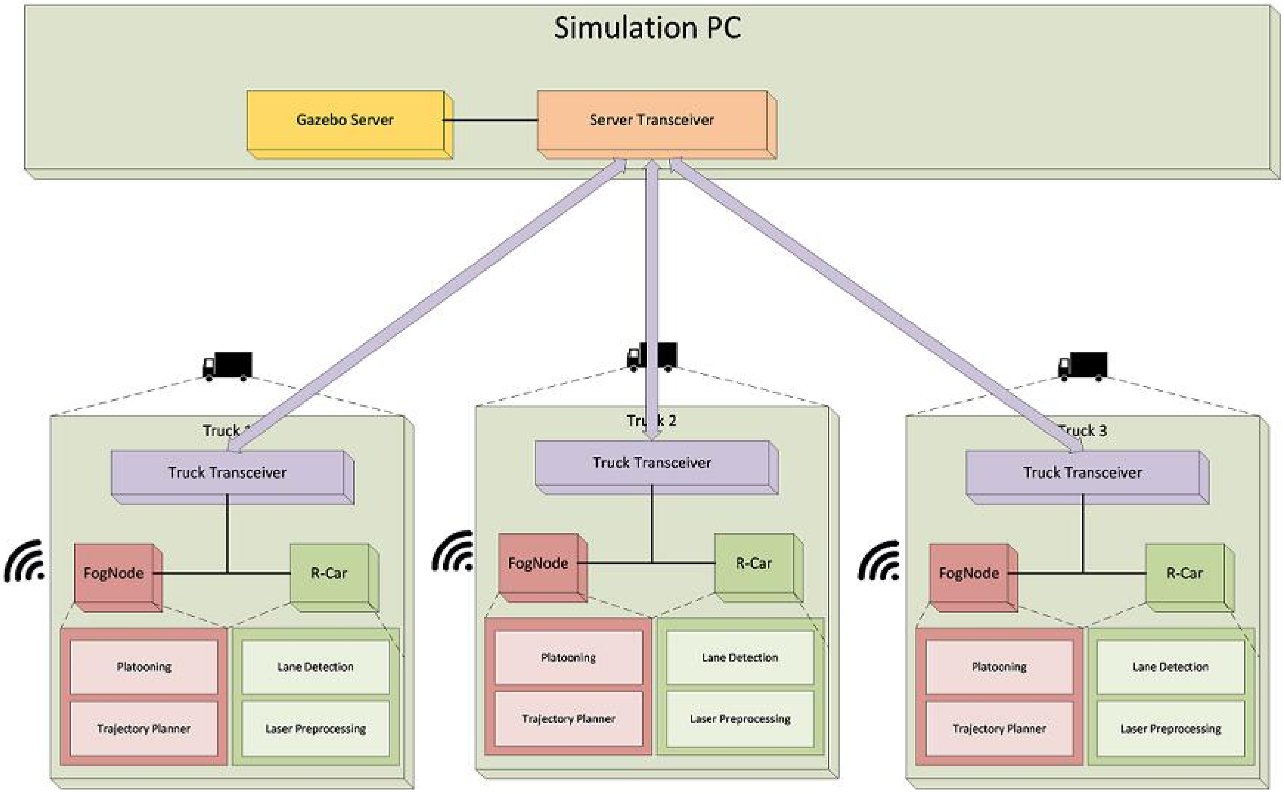


Figure 25: Simulation Architecture

Finally, three trucks were deployed in the lab demonstrator with the proposed architecture combined with the three simulated trucks. The sensors and actuators will be simulated as well. They were equipped with live sensor data and controls from/to the simulation tool (Gazebo).

7 Safety

ISO 26262 is an international standard for functional safety of electrical and electronic systems in production automobiles. Functional safety features form an integral part of each automotive product development phase, ranging from the specification, to design, implementation, integration, verification, validation, and production release. It defines functional safety for automotive equipment applicable throughout the lifecycle of all automotive electronic and electrical safety-related systems.

TTTech follows this standard when developing series automotive components. In the case of CPSwarm, only safety-compliant components such as R-Car ECU which is ISO26262-compliant are used. Although systems developed in CPSwarm are prototypes and developing them according to complex safety standards would not be economically feasible, such requirements will be respected during carry-on developments and exploitation planning.

8 Final steps conducted to completion

Concerning the automotive use-case, the final steps conducted during the last work period were to complete the design of the deterministic WLAN development. In parallel, the integration of the individual components of the demonstration was finalized successfully. In first instance, the deterministic WLAN connection was implemented by a standard LAN wired connection in the lab set-up. As soon as the deterministic WLAN was readily designed and available for integration, it was used to replace the LAN connection. Finally, the entire set-up was subjected to testing and the verification and validation process.

8.1 Optimization/Extensions

Certainly, it is not possible to exhaustively develop all kinds of traffic scenarios for the simulation tool within the course of a project of the size and dimension such as CPSwarm. Thus, the following list may provide some evidence for how the simulation might be extended /optimized in its carry-on development once CPSwarm is finished.

- Drive at lane X.
- Emergency Breaking.
- Change Cruising Speed.
- Change Max Speed.
- Shortest path algorithm.
- Dynamic selection of the role (leading/follower vehicle).

8.2 Abstraction library

The abstraction library can interface the wireless deterministic driver via a specific API. This is the major input into the abstraction library implemented in the automotive use case. Beyond this, the traffic simulation is envisaged as a potential contribution.

9 Conclusion

The document provides an overview on the finally conducted activities in the CPSwarm Project T8.3 with respect to the automotive use case obligations. In particular, it explains the elements designed and developed within the frame of the CPSwarm T8.3 activities. The major insight and achievement gained was the break-through development in the area of wireless deterministic data communication link that will enable designers to also make use of automotive grade wireless links in case of safety relevant applications. This was demonstrated in an example demonstration using a platooning use case application that has been designed in a simplified implementation suited to show the needs of wireless communication in automated driving scenarios.

References

[1] Isochronous Wireless LAN for Real-time Communication in Industrial Automation (ebook, see https://www.springer.com/de/book/9783662491577)

Acronyms

Acronym	Explanation
BE	Best Effort
CPS	Cyber Physical Systems
DG	Directorate General
ECU	Electronic Control Unit
LAN	Local Area Network
RC	Rate Constraint
RTD	Research and Technology Development
SUMO	Simulation of Urban Mobility
TTEthernet	Time Triggered Ethernet
TDMA	Time Division Multiple Access
TRL	Technology Readiness Level
TT	Time-Triggered
WL	Wireless
WLAN	Wireless Local Area Network

List of figures

Annex 1: The Role of Fog Computing in the Future of the Automobile

While today's cars manage to balance well some factors (economic efficiency, environmental sustainability, safety, and passenger comfort) automotive electronics face other challenges in the future: (i) higher density of functions per ECU, (ii) improved communication, inside the car as well as outside, (iii) security and privacy, and (iv) advanced processing needs for autonomous vehicles. These four challenges are complex. The response to these challenges, shaping the future of the automobile, is articulated around two main themes:

- Fog computing;
- Time-Triggered technologies, based on precise time distribution, time-sensitive networking and computing resource allocation making up a collection of design patterns, applied in critical computer-based systems.

Fog computing, with its "Edge of the Network" positioning and more constrained resources, extends cloud computing in a non-trivial way by introducing:

- Hard real-time and more deterministic behavior in its networking, computing and storage;
- Focus on the direct support of a much wider set of networking technologies, including wireless and sensor networking, as well as legacy wired networking;
- Relevance of mobility;
- Focus on the interoperability with non-homogeneous data sources, creating a data mediation functionality enabling applications to have more agile and flexible access to a wide variety of data sources;
- Support of compact, streaming, and real time capable data analytics;
- Extended system, networking and physical security and safety;
- Renewed interest in hardware support of functionality, motivated by energy, space and real time requirements.

In future vehicles, more sensors, connected via wires or, in higher numbers, wirelessly, will collect and report more sophisticated information. Video, laser and radar technologies will be key in the support of assisted drive. More microphones will help in preventive maintenance, voice activated control, and sound management. Driver health sensors will monitor key vital parameters.

Communications with other vehicles and the infrastructure will see the full adoption of WiFi DSRC with its use for both collision avoidance and general meshed vehicular communications. Multiple cellular connections, including new long range, low power connections, will be pervasive. The vehicle cabin will become an entertainment and information center, as well as a mobile office, served by WiFi, Bluetooth, NFC, low power sensor networks, with high bandwidth available for video, voice and data over IP. A rich computing and storage capability will be required to support high quality experiences in music, video and gaming. Networking will move more in the direction of IPv6.

In vehicle storage requirements will continue to grow. More data, even "Big Data", will be collected on both the vehicle health and on the passengers' health and experience. Some of this data will need to be processed, compressed or analyzed in real time on the vehicle, and some will need to be uploaded towards Data Centers and Clouds. Large amounts of navigation, entertainment, control data, and software will be downloaded into the vehicle.

Naturally, driven by the evolution of smart phones, the automobile will need to become a platform for the delivery of applications and become more open to the judicious use of Open Source Software.
All the trends above point in the natural direction toward Fog computing.

Fog computing provides scalable computing and storage architecture, rich wired and wireless connectivity support, virtualization for both non-real-time and real-time services, sophisticated data management and analytics support, secure computing and networking, and modern management and application deployment features.