D8.5 - INITIAL AUTOMOTIVE DEMONSTRATION

Deliverable ID	**D8.5**
Deliverable Title	**Initial Automotive demonstration**
Work Package	**WP8 – Use Cases Implementation**

Dissemination Level	**PUBLIC**

Version	**1.0**
Date	**16.07.2019**
Status	**final**

Lead Editor	**TTA**
Main Contributors	**Artiza Elosegui, Andreas Eckel (TTA,TTTECH), Melanie Schranz (LAKE)**

Published by the CPSwarm Consortium

Document History

Version	Date	Author(s)	Description
0.1	31.12.2018	Artiza Elosegui Andreas Eckel (TTA, TTTECH), Melanie Schranz (LAKE)	Initial text
0.2	29.01.2019	Artiza Elosegui Andreas Eckel (TTA, TTTECH)	Implementation of review findings
1.0	16.07.2019	Andreas Eckel (TTA, TTTECH)	Further updates inserted

Internal Review History

Review Date	Reviewer	Summary of Comments
15.01.2019	Alessandra Bagnato (SOFTEAM)	Comments on deliverable content
18.01.2019	Etienne Brosse (SOFTEAM)	Few comments to be addressed
05.02.2019	Gianluca Prato (LINKS)	Additional comments on structure and content

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 to 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 focus on three scenarios related to:

a) Swarms of drones and ground robots;

b) Swarm Logistics scenario;

c) Automotive use case.

This Deliverable "**D8.5 - Initial Automotive demonstration**" describes the work carried out in "Task 8.3 Automotive use case". The consecutive results until M36 will be reported in "D8.6 – Final Automotive demonstration".

This document provides first information on the planned demonstrations and on how the developments carried out in this domain will be 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:

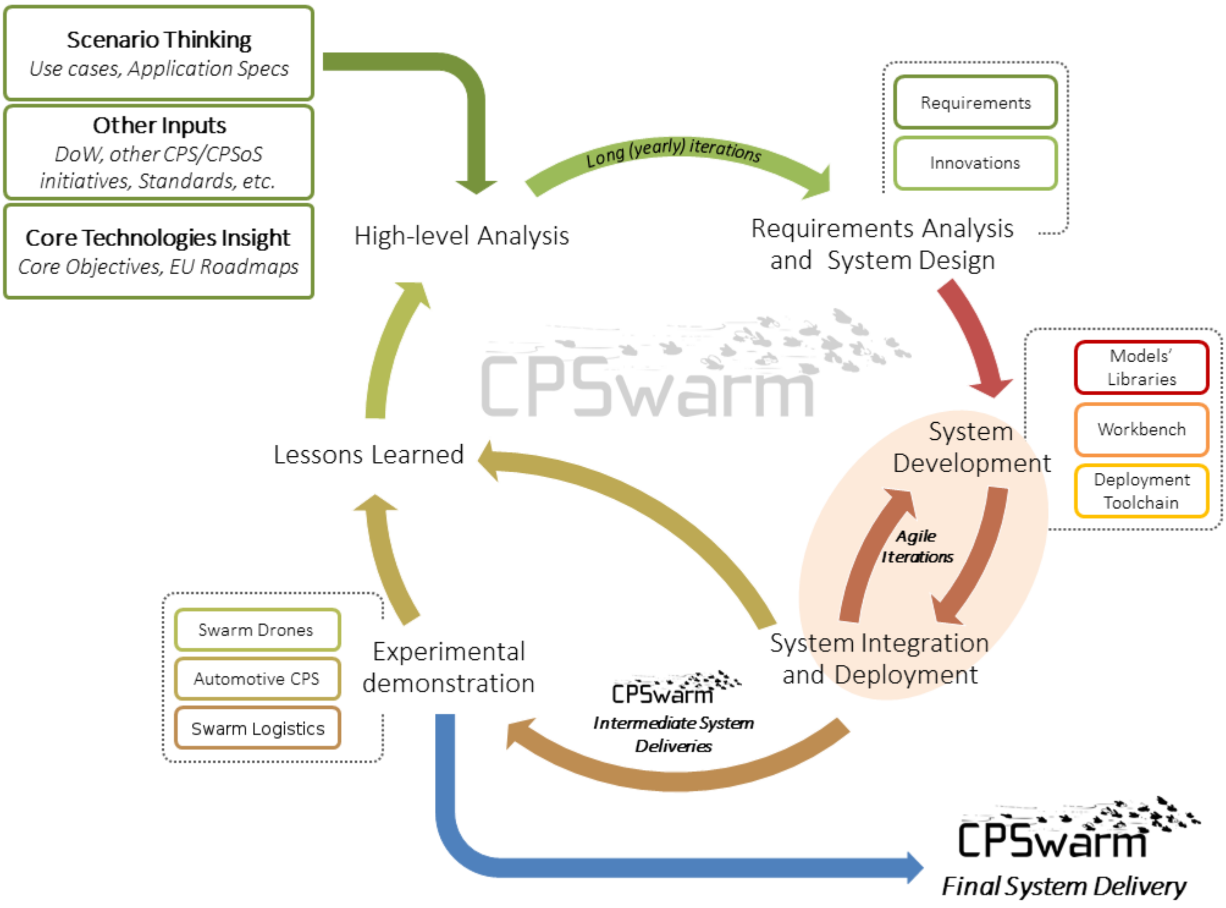


Figure 1: CPSwarm project lifecycle.

The "**D8.5 – Initial Automotive demonstration**" document is a public deliverable focused on the results of *Task 8.3* Automotive use case *scenario* at M24 of the project.

This deliverable is the result of the "Task 8.3 Automotive use case" and provides the first descriptions on the automotive use case. Another deliverable called "D8.6 Final Automotive demonstration" will be released in M36 describing the final version of the use case and the implemented features.

TTTech is the T8.3 leader and responsible for the delivery of D8.5.

2.1 Document organization

The document is organized as follows:

Final Vision: The document first of all provides an initial, first version (The "final Vision" will be laid down in the successor document D8.6) to explain the target application scenario of the development planned. This does not emphasize that all these features described will actually be implemented within the frame of this project. The extent the project will cover developments is described here-in.

Integrated Behavior: this contains detailed descriptions about the platoon use case.
State Machine: Contains the description of the state machine for the simulation.
Deterministic Wireless (WL) Driver: Contains the block diagram/description of the WL driver developed in this Use Case.
Laboratory Level Demonstrator: Contains the description of the proposed demonstrations
Simulation Framework: Contains the description of the Simulation Framework

2.2 Related documents

ID	Title	Reference	Version	Date
[RD1]	Final Vision Scenarios and Use Case Definition	D2.2	1.0	M16

3 Automotive use case (Platooning) architecture

The architectural set-up of the use case will serve as a basis for all further considerations. Based on Figure 2, it will consist of four elements:

a) The autonomous driving computer handling the environmental awareness capable of handling the smart sensors and control of the smart actuators as a reaction on the environmental awareness sensors.
b) The protection against external attack by building a trusted environment on board of each vehicle
c) The wireless data connection with deterministic behavior to connect to the other members of the platoon
d) The mission computer handling the different states of the platoon (i.e. different states: i.e. mission start, mission over, mission abort)

The architectural approach consists of two major parts:

a) The safety-relevant driving capability requiring real time behavior (i.e. all environmental awareness generation features and capabilities, the autonomous control of the vehicle including control of actuators etc.). It also includes the deterministic, wireless communication channel connecting the leading vehicle of the platoon with the following participants of the platoon.
b) The mission control part that is foreseen as a state machine-driven element isolated from the safety- and security- critical control data communication although integrated in the trusted data communication domain. The mission control computer will use the best effort communication link in the architecture and thus is separated from the autonomous driving related area running in a deterministic Ethernet channel of the network.

Each vehicle in the platoon will be equipped with the same capabilities whether realized by the same components or other providing the same functionality as a prerequisite for such application.

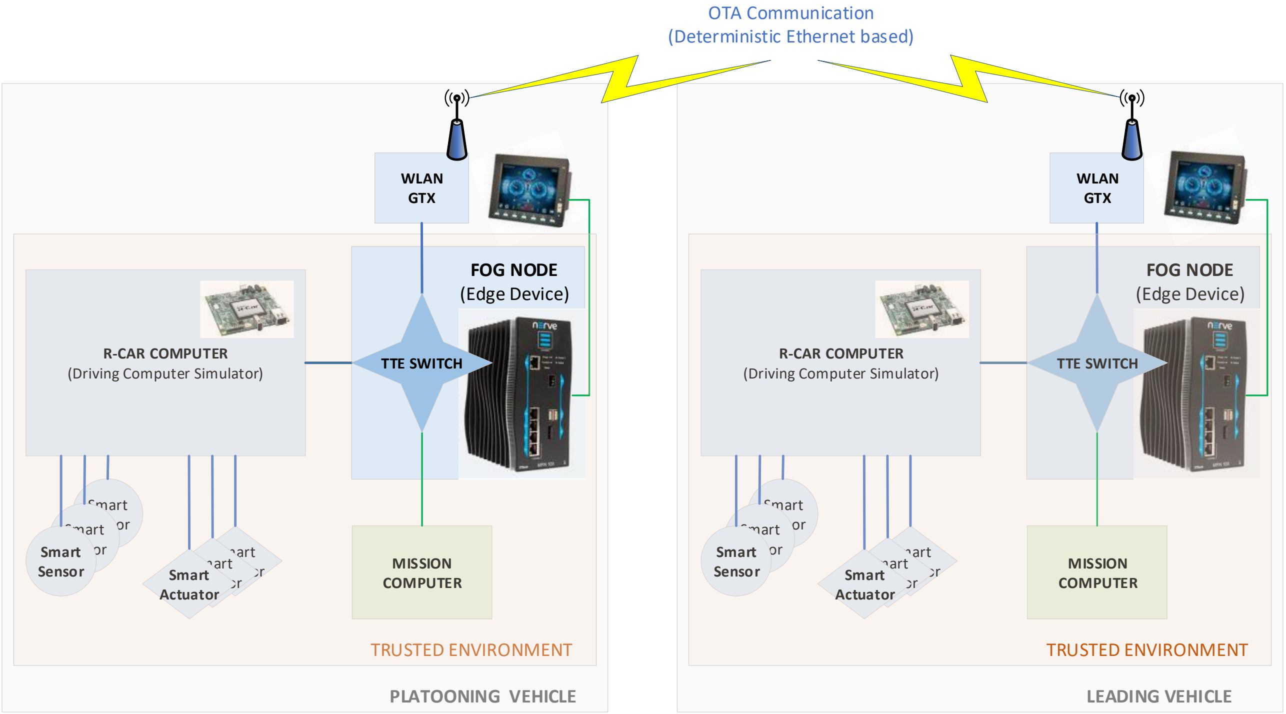


Figure 2: Architectural set-up of the automotive use case

“blue boxes/connections”: Deterministic Ethernet,
"green boxes / connections”: Best effort Ethernet traffic

4 Vision scenario

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

Vehicle Platooning Concept:

- The leading vehicle has autonomous driving capability 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 have autonomous driving capability and environmental awareness, too, to be able to react on specific driving scenarios requiring separate action (i.e. lane change and not enough space in new lane due to heavy traffic). In general, they follow the leading vehicle´s actions.

Figure 3: 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 (due to the opportunity to reduce significantly the distance between vehicles below the human response time requirements), as the trucks drive closer together at a constant speed, with less braking and accelerating. Consequently, freight vehicle platooning has also the potential to reduce the drag significantly resulting in less fuel burn and reduced CO2 emissions. Likewise, connected driving can help improve safety, as braking is automatic with virtually zero reaction time compared to human braking. Finally, platooning also optimizes transport by using roads more effectively, helping deliver goods faster and reducing traffic jams. These are some of the reasons, why platooning will also make sense in times with autonomous driving capabilities implemented in (freight-)vehicles.

Vision scenario in urban environment

As example, the following final vision scenario has been defined (although in practice the platooning configuration would most likely only be used in long-distance driving scenarios at higher speed rather than in cities). However, there are application scenarios that might even make sense in urban environments. This can as an example refer to "mixed" traffic, where potential platoon participants can be trucks, passenger-vehicles or even busses. The results can widely be transferred to long distance routes as well.

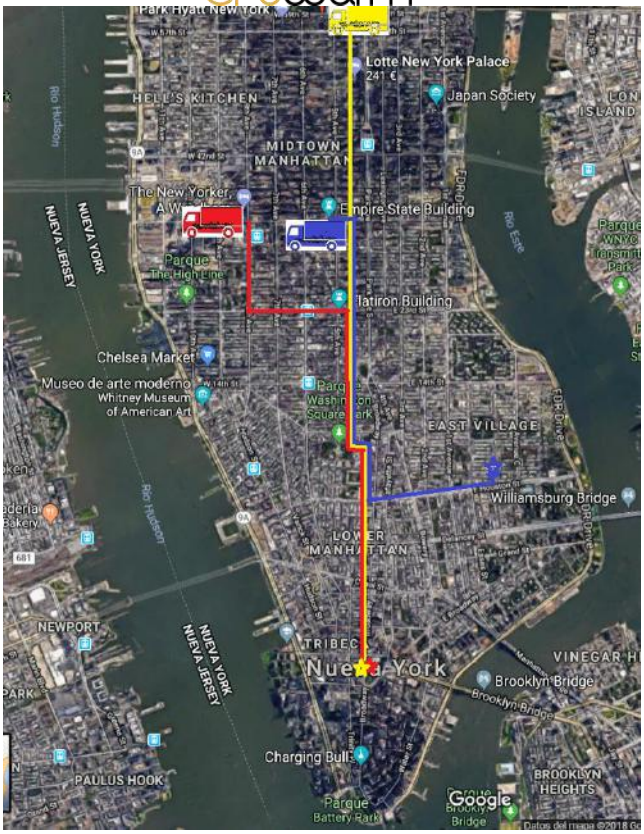


Figure 4: Example of 3 vehicles in a platoon

Referring to above figure, 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 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 final destination.

Advantage compared to individual driving (autonomously or not): in case a local driver is behind the wheel of the leading vehicle, it will use the most effective connection and thus be faster in reaching the goal since such local know-how can include construction, accidents, rush-hour traffic, garbage truck ahead, etc. much better than an autonomous route planner might do even if supported by local internet based support services. As an example, I can state the Flixbus from Vienna to Graz, if I take my navigation system it takes me 10 to 15 minutes longer to get to the city center than the route chosen by the Flixbus drivers since this is optimized. Thus, in case I am e.g. at Graz Murpark on my way to Graz city center, following a Flixbus helped me a lot!

Vision scenario in long-distance journeys

This is the classical application, where advantages become directly visible. Consider a bus driving from Vienna to Brussels (e.g. representing the yellow vehicle), a truck going from Bratislava to Frankfurt (representing the red vehicle) and another truck going from Budapest to Munich (representing the blue vehicle) and of course assuming that they would be in reach at their potential joining waypoint, the advantage becomes obvious. The blue vehicle reaches Vienna 20 minutes before the bus (yellow vehicle) leaves from Vienna to Brussels. The driver was active for 3 hours including waiting times at the borders. So, he decides to make a short break at a motorway service area to drink a coffee. All will use the motorway A1 for a longer distance. The truck representing the red vehicle is close to the other truck at the time of his departure from the service area. The truck from Budapest to Munich decides to go into a platoon with the bus that just is passing the highway entrance of the motorway service area. The Truck Bratislava Frankfurt realizes that and decides to join the platoon just started. The mission computer on the leading bus communicates with the two trucks following in the platoon. The mission computer of the truck to Munich (blue) will follow the platoon until the junction of A1 and A8 (which branches off shortly after the city of Linz to Passau and then continues either in another platoon or autonomously or directly driven by human driver. The truck to Frankfurt will stay in platoon configuration until the exit to Frankfurt and the bus probably will continue separately alike the truck to Munich.

Advantage compared to individual driving (autonomously or not): mainly the reduction in fuel burn resulting in lower cost and less emissions plus the fact that the platooning vehicle drivers may even do completely other work in their cockpits that might add value to their company/organization. It may also mean that they can proceed despite they would other have to stop for a resting period and thus now are able to efficiently continue without being on the wheel (similarly as in autonomous mode but at higher efficiency).

4.1 Integrated Behaviors

Although we are providing a lab demonstrator within the goals of the project only, the final vision for the vehicles have full autonomous driving capability thus they are able to take decisions when, for example, an obstacle on the route does not allow them to continue with their route within the platoon. As an example, they can change the lane (drive at left lane behavior) or brake to a full stop (emergency braking behavior).

Some vehicles might create a platoon 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 and as set in the mission computer.

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 data information. The vehicles are connected via deterministic wireless data link (explained later) when they run in a platoon so that the leading vehicle can prescribe actions and decisions (i.e. navigation, decision on take-over maneuvers, sequencing maneuvers, lane change, braking, accelerating etc.) for the follow-up vehicles. Relevant properties of such a distributed automotive system can be modelled supported by the CPSwarm workbench.

When part of the route is common for two or more vehicles, they can create a platoon responding to the swarm intelligence (i.e. less energy required for the followers). When they run in a platoon, the leading vehicle has autonomous driving capability 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 have autonomous driving capability and environmental awareness in case the platoon needs to be dissolved due to a specific traffic scenario. They follow the leading vehicle´s actions as long as they are a member of the platoon.

The mission describes the goal of the vehicle, i.e., which is its current position and where it should go. The optimization requested will be done on the route needed to execute the mission with the lowest cost. When two or more vehicles are traveling behind each other the cost of the road is reduced by 20% making platooning a preferred solution. In addition, fuel burn savings will reduce cost and emissions leading to economic and environmental advantages. The goal of the optimization will be to find out the best route for every vehicle. Therefore, the behaviors will be:

- The shortest path or shortest time or lowest cost etc. algorithm for each vehicle, from start position to final destination, 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. The implementation of such criteria will be conducted in the mission computer.

Such algorithms were developed for this application to show the benefit of suitable optimized "Swarm behavior" (LAKE).

Some vehicles might create a platoon 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 (in the mission computer).
When they run in a platoon, the vehicles will be running with a given speed and with a given distance between them. The controllers 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 able to take decisions and the behaviors will be:

- 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.

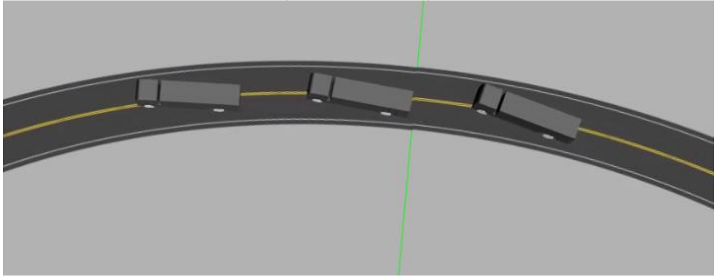


Figure 5: Platoon configuration

- Situation 2: Emergency breaking.

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

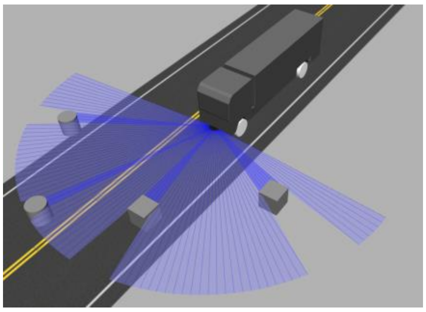


Figure 6: Obstacles detection for emergency breaking

4.2 State machine

Based on the vision scenario described above, the following state machine has been designed in collaboration with LAKE.

Threads:

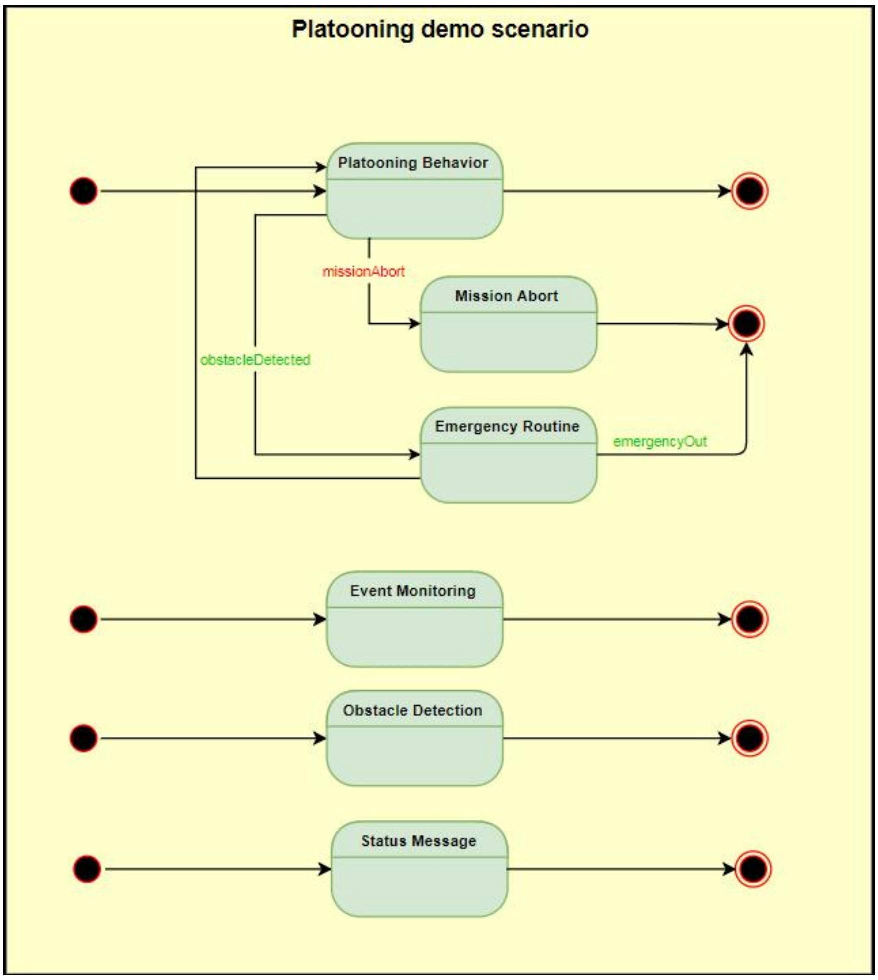
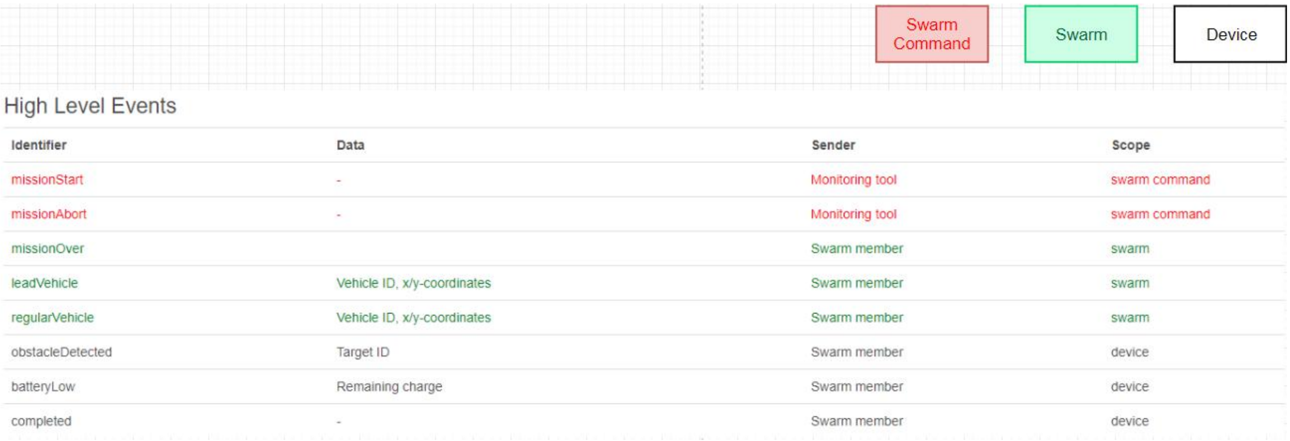


Figure 7: State Machine Demo, Model Scenario

1st level:



High Level Events

Identifier	Data	Sender	Scope
missionStart	-	Monitoring tool	swarm command
missionAbort	-	Monitoring tool	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 charge	Swarm member	device
completed	-	Swarm member	device

Figure 8: 1st Level

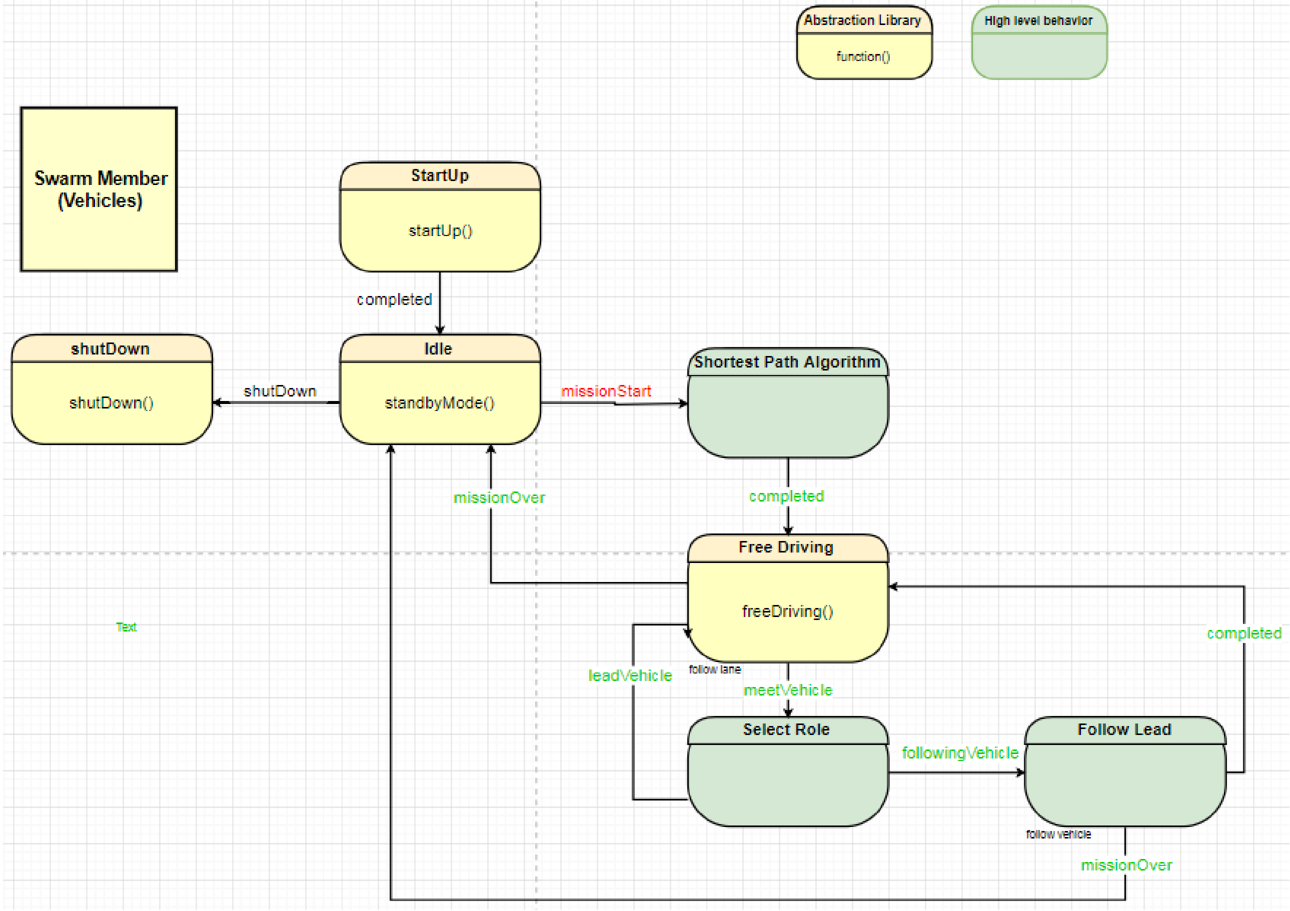


Figure 9: 1st Level Demo Model

2nd level:

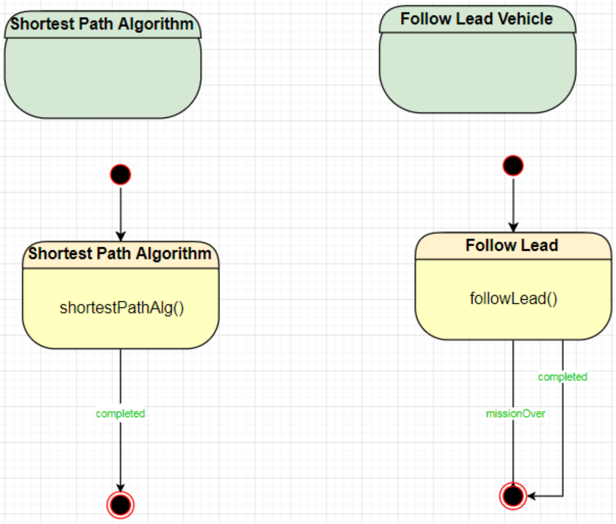


Figure 10: 2nd Level Demo Model

Emergency routine:

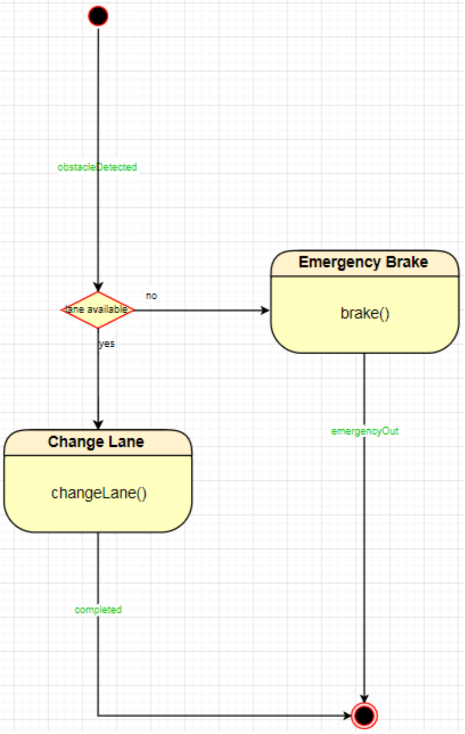


Figure 11: Emergency Routine Model

The state machine will be implemented as a "Layer" on top of the safety-relevant, time-triggered control data communication.

5 Deterministic wireless driver

Autonomous vehicles can only communicate with each other over the air (wireless) while they run on the road. The challenge therefore is to apply the know-how of the wired Deterministic/TTEthernet on a wireless environment. Deterministic /TTEthernet is a scalable technology and allows development of critical system parts according to fail-safe or fail-operational application requirements.

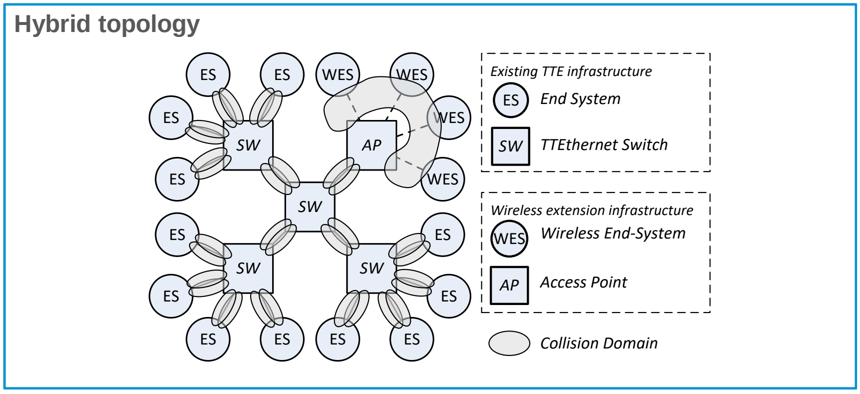


Figure 12: TTEthernet topology

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

In order to support integration of applications with different real-time and safety requirements in a single network, Deterministic/TTEthernet supports three different traffic classes:

- time-triggered (TT) traffic - is sent in a time-triggered way, i.e. each Deterministic/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 (not used in the demonstrator, mainly implemented for aerospace applications.
- best-effort (BE) traffic - traffic with no timing guarantees. BE traffic class compatible with the IEEE 802.3 standard Ethernet traffic (will be used for the mission computer data communication).

Challenges of the automotive scenario

1. Wireless communication

   The communication from the leading vehicle to the follower vehicles, and also among all platoon participating vehicles as well as those intending to join the platoon, must mandatorily be wireless since it is not possible to have a wire among vehicles when they are running in a realistic situation.

2. Real-Time communication

   Real-time communication is compulsory for all safety/security related data communication (i.e. in the use case all autonomous driving related communication) to give response to the safety requirements, for example, when breaking. Network communication technology must use time scheduling to implement deterministic real-time communication.

3. Low reliability communication

   Real circumstances 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.

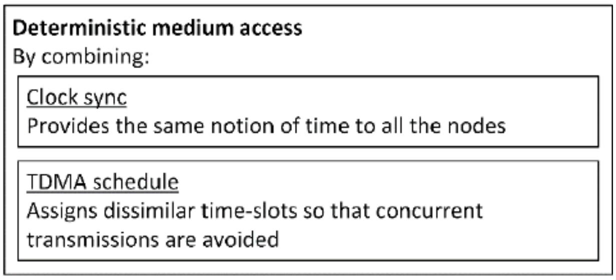
Wireless Driver integration

The wireless driver developed in CPSwarm meets all the requirements of the car platooning scenario mentioned above. The driver will be 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 data transmissions between vehicles consists on scheduling the points in time when every vehicle is able 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 do not happen. For the schedule to be followed properly, a common time notion should exist between the vehicles. To solve this issue, a time synchronization protocol between the vehicles was be deployed.

The deterministic WLAN implementation is based on the combination of two concepts:

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 13: Media access with deterministic behavior

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 is currently ongoing in its final steps (integration tests under performance). The API can either include normal sockets and addition configuration through standard driver calls, or alternatively, it will use custom API that wraps the sockets in the same way as the TTEthernet. Both options have 2 different traffic classes: (a) best effort and (b) time-triggered.

Devices used for the driver testing:

Figure 14: Test Devices

It is certainly understood that in a product-ready module for platooning the involved control electronics and algorithms would have to fulfil safety and security requirements that 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 will exceed the means of the CPSwarm project.

6 Laboratory level demonstrator

The automotive use case will be implemented by a laboratory level demonstrator (TRL 3/4, demonstration in breadboard lab environment) around autonomous driving vehicles equipment connected via electronic drawbar (platooning). Since the electronic platforms are mainly used for intra-vehicle computation (inside vehicle, wire based), the aim of the demo is to allow safety-related inter-vehicle communication (OTA communication) to enable coordinated actions such as the ones described in the scenario. To do this, a wireless connection suitable for safety-related data communication (TTEthernet based) among the vehicle computing platforms has been developed.

The automotive demonstrator is based on the architectural block diagram provided in Figure 2. Smart sensor data will be generated off-line and feed to driving computer which will send command/control data to simulated, smart actuators visualized on the screen of the set-up. The main goal is to show the OTA link and its suitability in extending wired communication on the different vehicles by wireless data connection supporting safety-related data communication.

The idea of the fog node is to add computational power and data processing in the vehicle and “offer” this real-time processing power to other units in the vehicle, as well as implementing the “trusted environment” to protect against attacks from the cloud. Furthermore, the Fog Node can preprocess data to reduce the amount of data that needs to be transmitted prior to transmission to the cloud. The fog node acts thus also as a gateway between the Cloud and other end devices (ECU).
The demo will consist of three 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 will not contain a full platooning application software since designing such application is not the expertise of the consortium partners. The demo will focus on the data communication.
c) In addition to the initial DoA, it will also include an implementation of the state-machine based mission computer supported equipment. This part of the software is in general generated by the OEM or the Tier one supplier and is not a direct competence of TTTech. However, since it has been detected that this implementation is not covered by the DoA, TTTech has decided to cover a baseline dummy implementation in order to demonstrate the feasibility of the State-machine approach provided by Lake.

6.1 Hardware

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

MFN 100
Edge Computing Device

Figure 15: TTTech Fog node

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

R-CAR

Figure 16:R-CAR Device

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

The Mission Computer

We currently check for best suitability: We will either use a simple "Beagle-Bone" Computer or the TTControl GmbH Computer TTC580.

Display

We will either simply connect a standard table computer screen, or we might connect the TTControl GmbH Display used for "off-highway domain, the HY-eVision^2 10.4. The decision will be made during final demonstrator set-up.

6.2 Architecture

The architecture example shown in Figure 17 is respecting typical distribution of functions as generally applied in the automotive industry. Related to our approach also displayed in Figure 2 such functions and related application software may be distributed between the Fog node and the R-CAR node.
. 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.

The proposed architecture is:

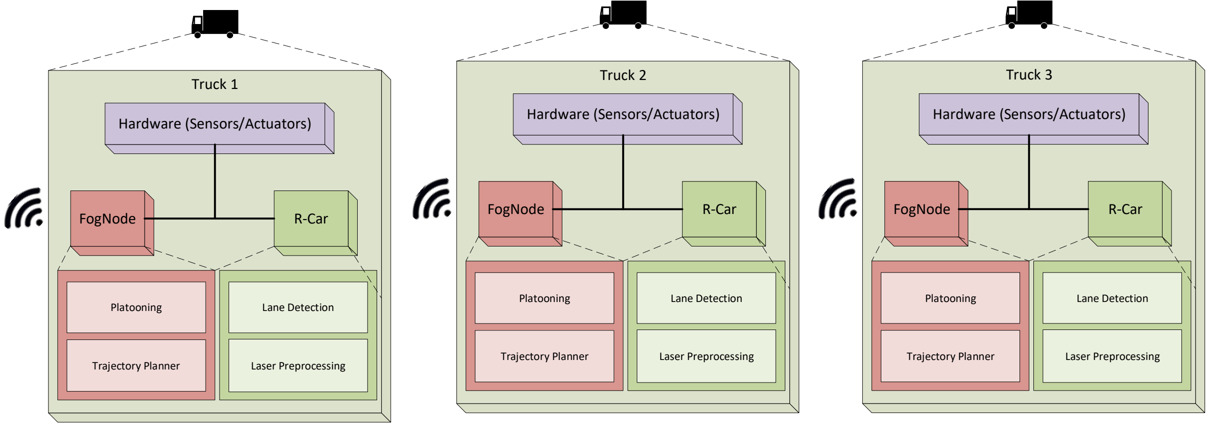


Figure 17: Architecture of the automotive use case

Each of the vehicles is considered a black box and is responsible for each own sensors and actuators. Only the FogNode is visible from outside (any communication from externals goes through it).

The vehicles, through the FogNodes and the appropriate interfaces for the safe deterministic wireless link, can communicate with each other via wireless even for safety-related data communication. 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. Referring to Figure 18 the architecture for the platooning use case can easily be integrated into a standard components-based architectural scheme.

The list of components often found in modern vehicles and what will run where is shown below:

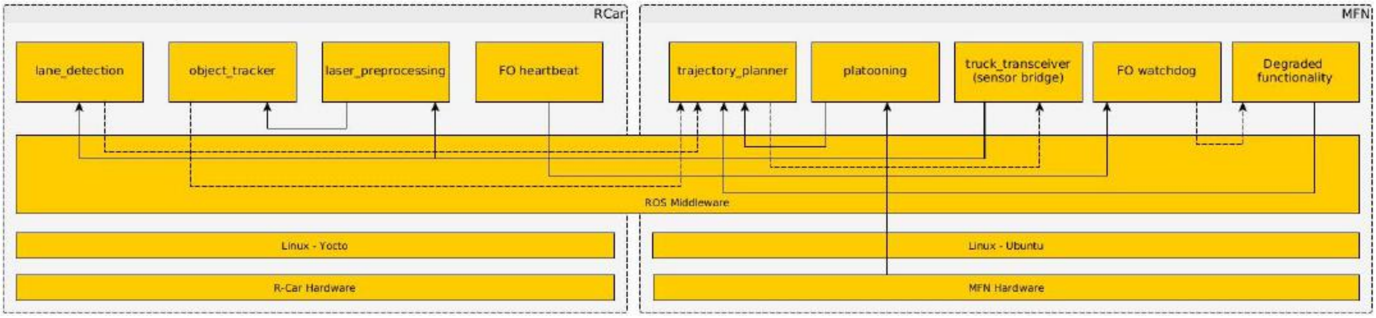


Figure 18: Components List

6.3 ROS node/ROS Application for the Automotive Use Case

Initially we assumed that we could integrate ROS into our safe & secure deterministic Ethernet data communication architecture. Despite significant effort invested according to our large interest to be successful with this approach we need to conclude that the attempt to integrate ROS into the deterministic, schedule based deterministic Ethernet concept was a failure. It proved to be impossible to merge with the real-time, schedule-based deterministic concepts of Deterministic Ethernet.

Initially we assumed that the ROS node could either be integrated in the FogNodes or the R-Car node. We expected that this would result in a set-up closer to a product approach. Within CPSwarm we suggest an architectural concept, which would have to be discussed with a potential customer. Checking this in our customer base we also did not find any kind of agreement for the suggestion even the other occurred that discussion partners showed a strong disagreement for such suggestion. Thus, we decided to cancel this approach and stay with our direct approach with Deterministic Ethernet (not diluted by the ROS implementation nor accepting any compromise in performance required to achieve this).

6.4 Failover (FO)

A failover mechanism can also be installed if necessary, so that in case the R-Car fails the MFN can start up as an "emergency replace".

For that purpose, a heartbeat component and a watch-dog would be added to the system (in general anyway included in such safety related architecture). The FO heartbeat is a component sending a heartbeat to the watchdog. If the heartbeat didn't arrive (because i.e. the R-Car died) the watchdog will boot up some simpler software components in order to replace the applications running on the R-Car.

7 Simulation framework

7.1 Network simulation

A cost effort constrained way to test whether the communication between vehicles will 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++ discrete event simulator, in combination with the INET framework, allows for data communications network simulation. The task partners (in particular TTT) are working on 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 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., % 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 investigations aimed at showing the difference between a plain wireless network and a time-triggered network concerning the packet loss due to interference and collisions. When we assume a swarm of agents that need to communicate with each other or with a central station through wireless you need to consider the packet loss due to these 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, this results in 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 in order 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 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 starvation. 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).

Demo explanation

In the first demo a network with 6 agents named (A to F) is simulated. The northern agents (A, B) communicate with the southern agents (E, F). Due to the distance the last 2 agents act as relays between them. The demo 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 (in order 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 demo 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 doesn't go for the relays at the middle, where both of them are affected by the transmission of the two groups resulting in the agent not dropping almost all of the packets.

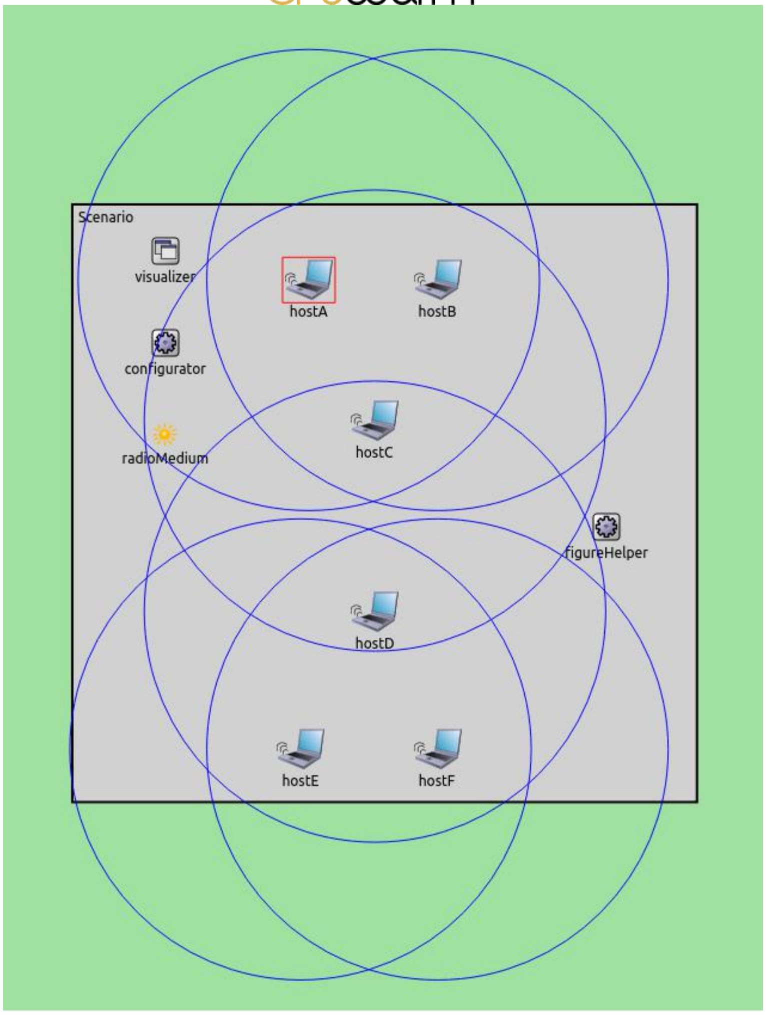


Figure 19: Demo 1 six Agents Scenario

As already mentioned, this was an extreme example were the traffic was so heavy resulting in this heavy packet loss. There are many ways around 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 didn't receive the update. So, it 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, you have a platoon of 10 trucks driving behind each other fairly close to each other to reduce 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 person 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.

7.2 Traffic simulation

This work comprised investigations on which simulation tool would be most suitable. 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, in order 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 was taken. SUMO (Simulation of Urban Mobility) is an open source road traffic simulation package designed to handle large road networks and is licensed under the GPL.

Figure 20: 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
- 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)
- Fuel consumption calculation

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

- OMNeT++ is a discret event Simulator
- The INET framework is a model suite for wired, wireless and mobile networks
- Currently used by Pablo 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 is an extension to VEINS and SUMO which adds platooning behavior to the simulation
- Permit is an extension to Plexe allows to simulate platooning maneuvers

Example: Platooning application on a highway

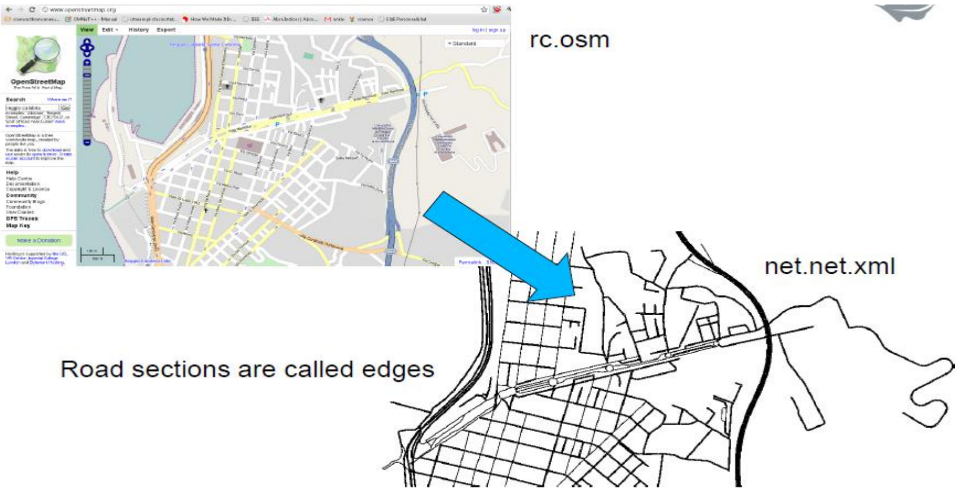


Figure 21: Platooning application on a highway

- 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*
-

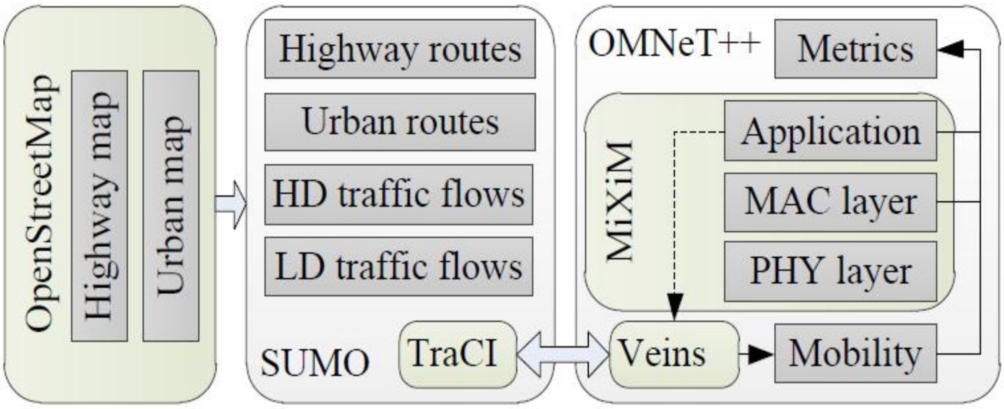


Figure 22: Platooning application on a highway - Model

8 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 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.

9 Final steps to completion

Concerning the automotive use-case the next steps will be to complete the design of the deterministic WLAN development and compete the integration and testing tasks for the UC Platoon demonstrator.
In addition we need to complete the design of the Mission Computer layer (state machine implementation) and integrate it on top of the safety-related deterministic control of the platoon including its driving computers with autonomous driving capability and the Fog node to result in a "trusted Area" for protection against unintended external access from any kind of cloud based malware application.

The work will be finalized by completing the test for the entire lab-based system.

9.1 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)

9.2 Abstraction library

The abstraction library connection with the automotive UC platoon application will be established via the state-machine layer. It will be deployed on top of the safety related autonomous driving capability and network.

Beyond this, the traffic simulation is envisaged as a potential contribution.

10 Conclusion

This deliverable presents the work done in Task 8.3
It provides an overview on the status and the finally planned activities that are outstanding until completion of the automotive use case obligations.

11 References

[1] Isochronous Wireless LAN for Real-time Communication in Industrial Automation

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.