EUROPEAN COMMISSION
SEVENTH FRAMEWORK PROGRAMME
Theme: ICT

Small or medium-scale focused research projects (STREP)

FP7-ICT-2013-10

Objective ICT-2013.6.5 Co-operative mobility

a) Supervised Automated Driving

GA No. 612035

Interoperable GCDC AutoMation Experience

Deliverable No.	i-GAME D3.2	
Deliverable Title	Proposal for extended message set for supervised automated driving	
Dissemination level	Public	
Written By	Jacco van de Sluis, Otto Baijer (TNO) Lei Chen, Hoai Hoang Bengtsson (VIKTORIA) Lorena Garcia-Sol, Pere Balaguer (IDIADA)	
Checked by	Aitor Ruano (IDIADA) Alex Voronov (VIKTORIA) Jan de Jongh (TNO)	23-09-2015
Approved by	Almie van Asten (TNO)	30-09-2015
Status	FINAL	30-09-2015

Please refer to this document as:

DEL_i-GAME_D3.2 Proposal for extended message set for supervised automated driving

Acknowledgement:

The author(s) would like to thank the partners IDIADA, TNO, TU/e and VIKTORIA for their valuable comments on previous drafts and for performing the quality assurance on the final draft.

Disclaimer:

i-GAME is co-funded by the European Commission, DG Research and Innovation, in the 7th Framework Programme. The contents of this publication is the sole responsibility of the project partners involved in the present activity and do not necessarily represent the view of the European Commission and its services nor of any of the other consortium partners.

Executive Summary

The objective of the FP7 i-GAME project is to facilitate development and real-life implementation of automated driving with a focus on cooperation, supported by wireless communication among vehicles and between vehicles and roadside equipment. To this end, the i-GAME project aims to develop vehicle automation approaches and interaction protocols utilizing wireless communication to cooperatively perform manoeuvres in both highway and urban settings, culminating in a real-life challenge of a Grand Cooperative Driving Challenge (GCDC) in which participants from both industry and academia are required to perform three cooperative scenarios with their automated test vehicle. So the participating teams and their vehicle implementation are judged in a competitive setting.

To support the interaction protocols used for the complex manouvering in these cooperative scenarios new message sets are defined. This document, being Deliverable 3.2 of the FP-7 i-GAME project, propose adaptations to current CAM (Cooperative Awareness Message) and DENM (Decentralized Environmental Notification Message) standards, and defines a new i-GAME Cooperative Lane Change message (iCLCM) that enable the cooperative automated driving needed for the GCDC in 2016.

This is done with an bottum-up approach, using knowledge from past experience during the Grand Cooperative Driving Challenge 2011, the predecessor of the i-Game challenge. And building on proven implementations, combining and extending these and using current available communication equipement and adapting this for the GCDC 2016 needs. The focus of GCDC is on interoperability, where V2V communications are based on ETSI standards on Cooperative Intelligent Transport Systems (C-ITS).

The message sets are constructed by analysing the previously defined interaction protocols for all the separate scenarios. The interactions needed in the main stages for merging, intersection crossing and emergency vehicle are used to derive message flow diagrams. From these message flow diagrams the information that needs to be exchanged can be determined and what the associated data types and characteristics are. From this the new messages sets are defined, taken into account the current available (CAM, DENM) messages, re-using, adapting and defining a new i-GAME message set. For practical implementations the message packet format and ASN.1 definitions are specified and available in the appendices. Finally the i-GAME Simulation Test Tool is introduced which will use the message sets and enable the teams to test their individual implementations.

Table of contents

1 Introduction

The i-GAME project

The objective of i-GAME is to develop technologies that speed-up the real-life implementation of automated driving, supported by communication between the vehicles. The objective is reached by enabling reliable interoperable communication in a multi-vendor, multi-network, multi-service environment in the context of a Grand Cooperative Driving Challenge (GCDC). The GCDC consist of different scenarios (highway and urban), in which the participating teams and their vehicle implementation are judged in a competitive setting.

The scenarios cover realistic use cases with consideration on the technology maturity and the near-future practical implementation. Interaction protocols are designed for each of the scenarios and will be used by the participants during the competition. The protocols define sequences of manoeuvres to perform the scenarios with the corresponding wireless message sets to enable interaction and cooperation. The interaction protocols are verified with simulations and real life implementations in a reference vehicle. With this unique bottom-up approach, separate implementations are combined and gradually enhanced. This approach includes solving interoperability issues and addressing the alignment of standardization. The basic idea behind the GCDC is that with the current evolution of Vehicle-to-Vehicle (V2V) communication technologies and the progress in standardization, the teams can equip their vehicle with communication equipment that is commercially available today. The focus of GCDC is on interoperability, where V2V communications are based on ETSI standards on Cooperative Intelligent Transport Systems (C-ITS).

Background to the Proposal for extended message set for supervised automated driving

GeoNetworking (GN) will be used for distributing the messages between the vehicles. This is an ad hoc routing protocol making use of the geographical positions for delivering the messages to the relevant vehicles in a certain geographical area. C-ITS introduces a facility layer to host different services for ITS applications. Two of the major services will be considered in i-GAME, i.e. the Cooperative Awareness (CA) that generates the Cooperative Awareness Message (CAM), and the Decentralized Environmental Notification (DEN) service that generates Decentralized Environmental Notification Message (DENM). The i-GAME communication architecture [1] will support both CAM and DENM. In addition, CAM and DENM are specified mostly for road and vehicle condition awareness, not for automated driving interactions, meaning that the current message sets should be extended and a new message set is designed to support the GCDC scenario interactions.

Each of the scenarios involves (complex) exchange of information by wireless communication. Detailed information flow diagrams are drafted up to illustrate the message flows of the interaction protocols as defined in [2]. The required information within each of the messages, needed for lower control to perform maneuverers, are identified and specified. First step is to define a basic message set that support the interaction protocols needed in the scenarios. Second is to utilize the available information within the current CAM and DENM standards as much as possible. Only when needed new message fields will be introduced. In designing the new messages we pursue alignment with relevant activities in other projects like AutoNet2030.

Chapter 2 identifies and describes the main stages in the three i-GAME scenarios. In Chapter 3 the scenario the message flows are constructed to determine the relevant information to be exchanged. Chapter 4 describes the information fields that are needed in the messages. This information is mapped with the current CAM and DENM or other relevant message definitions from other projects. In Chapter 5 Message packet formats are specified into detail. Chapter 6 focus on the ASN.1 format used in defining the message content. Tools are listed to compile the message set software libraries for the teams' own communication platform. The detailed message set information is presented in separate appendices. Chapter 7 describes the i-GAME Simulation Test Tool which allows participants to test (many of) the communication and controller aspects of their equipment well in advance of the actual event.

2 Interaction Protocol for the i-GAME scenarios

2.1 Introduction

The basic communication architecture and functional blocks are defined in i-GAME D1.3 Functional architecture document [3]. See Figure 1 for an overview of the i-GAME functional blocks and their relations. The top block are the defined scenarios for Cooperative Highway, Cooperative Intersection and Emergency Vehicle (EV). The first two are considered as part of the competition, whereas the EV one is for demonstration purposes only. Detailed information is available in i-GAME D1.1 [4].

From these functional blocks the needed requirements are derived to perform the manoeuvring in the scenarios: platooning, lane change, intersection passing. Then the interaction protocols are defined for the scenarios, as described in detail in i-GAME D2.1 [2]. These protocols define also the sequence of manoeuvres that are needed to perform a scenario.

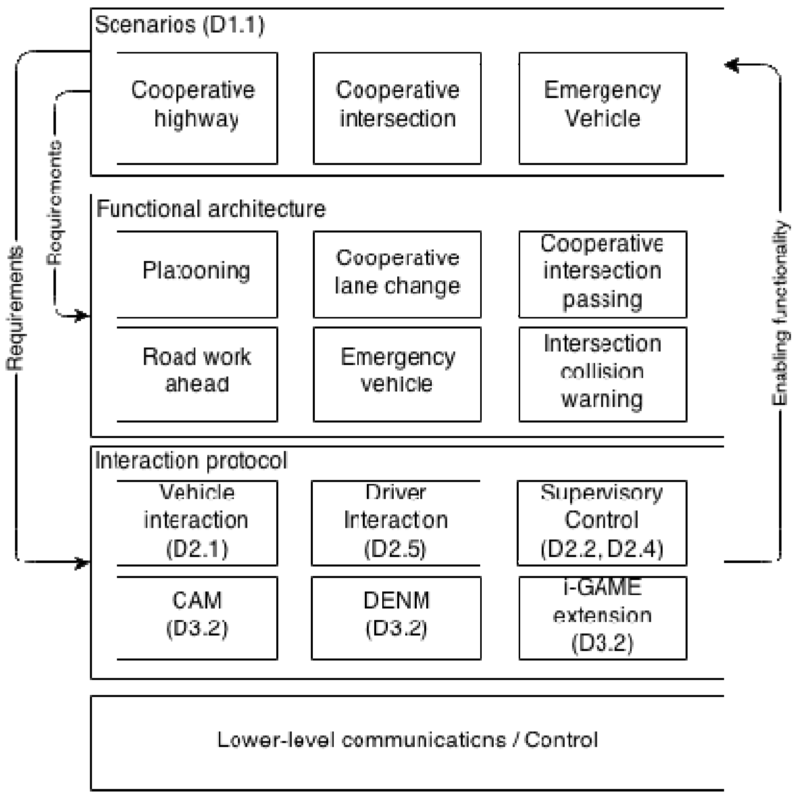

Figure 1 i-GAME functional blocks

From the interaction protocol, the information flow that is needed in the main stages of a scenario makes it clear what information needs to be exchanged. And what the content of the data fields in the message sets should be. In the Cooperative Intelligent Transport Systems (C-ITS) standards, the Facility layer is for information and application support. It provides the services and common functionalities to enable different ITS applications. For the execution of the scenarios the message definitions coming from the facilities on the Cooperative Awareness (CA), generating the Cooperative Awareness Message (CAM [5]) and Decentralized Environmental Notification (DEN), generating Decentralized Environmental Notification Message (DENM [6]). An additional dedicated message is defined set to support the complex manoeuvring in the i-GAME scenarios called the i-GAME Cooperative Lane Change Message (iCLCM).

CAM is the “Hello” message which is sent out frequently with information of the sending vehicle status. The purpose is to notify the surroundings about the existence of the vehicle with its relevant vehicle information. DENM is an (environmental) event triggered message, which is used to inform related vehicles of certain events, such as roadwork.

The lowest block of the diagram are the low-level communication requirements, in relation with the interaction protocol message sets, supporting the scenarios. The i-GAME communication architecture is based on ITS-G5 Vehicle-to-Vehicle (V2V) communication and must support CAM, DENM and the non-standard, iCLCM messages. The GeoNetworking standard (ETSI EN 302 636 series) will be used for distributing the messages between the vehicles. This ad hoc routing protocol makes use of the geographical positions for delivering the messages to the relevant vehicles in a certain geographical area. Detailed information for this is available in i-GAME D3.1 Wireless communication basic specification document [1].

2.2 Merging scenario

This scenario is based on the following starting points (see Figure 2 below):

1. A Road Side Unit (RSU) is being used for reproducibility and for managing the scenario execution. To make sure that the preconditions are the same when (re)starting the scenario. This phase is called the preparation phase, it is not part of a judging criteria unless the preconditions of the scenario are not met and further execution of the scenario is not meaningful.
2. “Stay in your lane”! It is not allowed to change lane unless you are in the actual merging process.
3. As soon as the platoons have a certain relative position to each other with the pre-defined speed, the scenario will start by sending the Road Works Warning message from the RSU. From a practical point of view this RSU signal will be send from an Organisation Pace Car (OPC) in lane A.
4. All vehicles belonging to the same platoon have the same speed upon entering the Competition Zone (CZ): Vehicles in lane A must synchronize their speeds with Vehicles in Lane B.
5. Platoon A is shorter than B.
6. Platooning is based on the following assumption: If a vehicle detects a vehicle in front, it will platoon with this vehicle in front. If there is no vehicle in front detected, the vehicle itself assumes it is the platoon leader.
7. From a safety point of view it is important that when communication and/or the algorithm stops the vehicle sends out an Emergency brake signal and stops. Vehicles receiving this message and positioned behind this vehicle should also immediately stop on receiving this message. Vehicles in front of the transmitting one do not stop but continue the scenario.
8. The OPCs do not have an active role during the scenario. They are included to:
    - Align and make sure the Platoons have the predefined starting speed and same position to each other to make the scenario repeatable.
    - Start the scenario by emulating a RSU send the Road works warning.
    - OPC A (and B for pairing) will, if necessary, participate in the merging scenario.

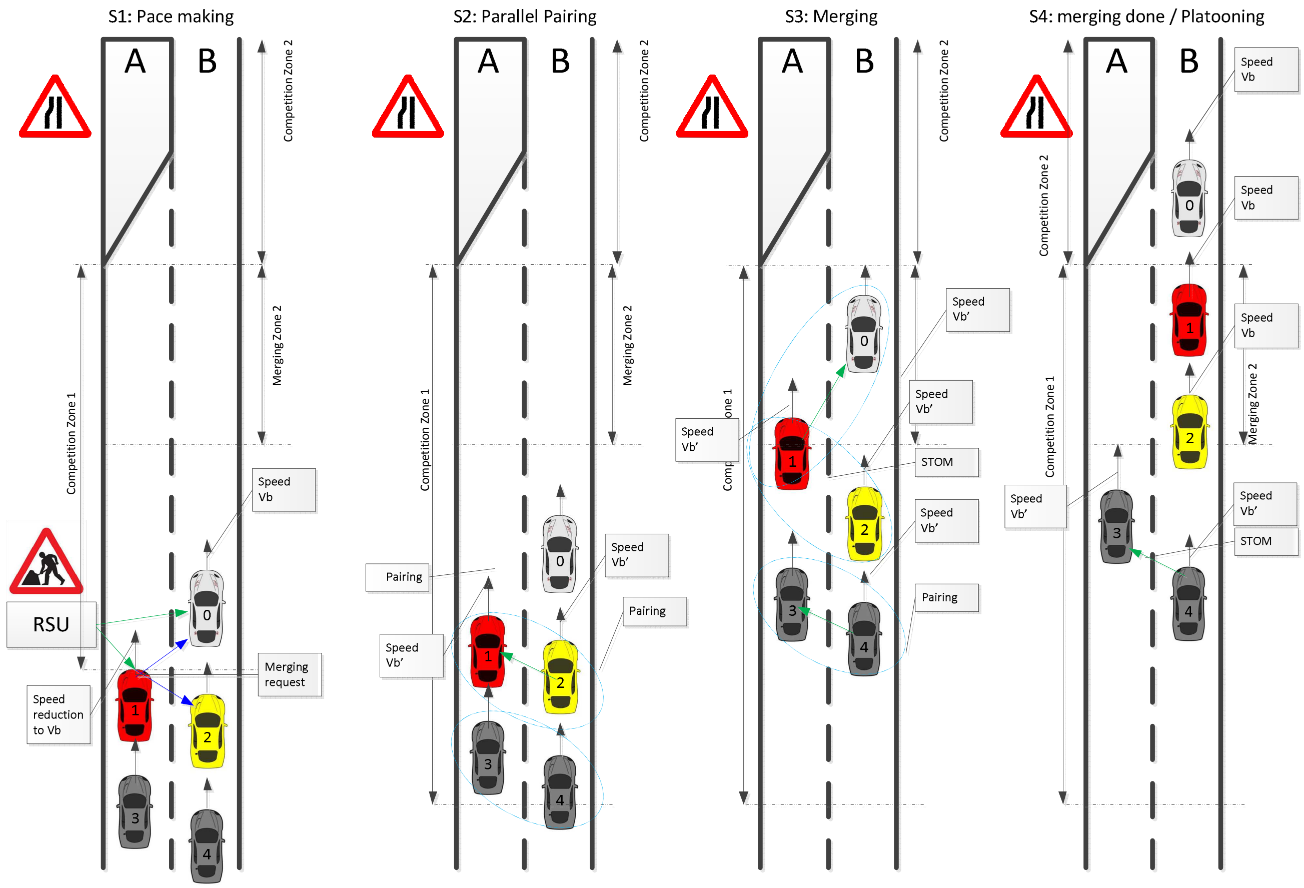

Figure 2 Merging scenario

Within the merging scenario there are 4 Stages distinguished that will be judged. There is also a preparation stage that will not be part of the judging criteria, this is defined for reproducibility of the scenario.

Stage 0: Preparation phase

- Participating vehicles start in their standby / wait state upon receiving CAM message from RSU, meaning: switch on the controller, choose the lane, choose the scenario. The first vehicle sets also the Lead flag and the role (OPC vs participant).
- RSU sends "start platoon B" message for platoon B / OPC B to start driving and accelerating towards a speed of 60km/h.
- After x (to be defined) seconds Platoon A, behind OPC A receives its "start platoon A" message and starts driving and accelerating towards a speed of 80km/h.
- After x+y (to be defined) seconds (just at the competition zone), OPC A and OPC B are at their initial starting positions, relative to each other in order to start the scenario. OPC A (simulating a road work RSU) sends a Road Works Warning and the competition starts. Please note that this alignment is the same starting position in order to make the scenario and the expected behaviour reproducible. This initial alignment is different from the alignment of the individual vehicles within the merging scenario starting in stage 1.

Stage 1: Pace making

By transmitting the Road Work Warning message the competition and the associating judging starts. The following actions are expected in this stage:

- Platoon A reduces its speed to 60km/h and aligns with Platoon B (pace making and position alignment).
- OPC Lane A/Vehicle_1 sends out merging request.
- Vehicle 0 is entering the Merging zone, by receiving this information from the RSU. Vehicle_0 does not see any vehicle in front of him, meaning there is no need for merging.

Stage 2: Parallel pairing (B2A)

This stage starts after reception of the merge request. Vehicle_1 and Vehicle_2 are also entering the merging Zone (see Figure 2 for vehicle indexing). Vehicle_1 detects vehicle_0 in front and Vehicle_2 detects Vehicle_1 and Vehicle_0 in front. Because of the rule "lane A merges in front of B" the following actions are taken:

- After sending and receiving the Merging request, platoon A and B reduce their speed to 40 km/h.
- Vehicles in platoon B: like Vehicle_2 and Vehicle_3 start pairing up with their forward most-important-object on left lane: (FWD MIO L) in above scenario: Vehicle_2 to Vehicle_1 and Vehicle_4 to Vehicle_3. The pairings may happen in parallel. Upon this pairing the vehicles in lane B start making a gap with respect to their pairing partner in Lane A.

Stage 3: Sequential pairing (A2B) and merging

- After platoon B has been paired up with A (after certain, to be decided, seconds from the start of completion) vehicles from platoon A start pairing up sequentially with Platoon B. Only the first vehicle in platoon A is allowed to pair with Platoon B. The selection of forward pairing partner for a vehicle in Platoon A is by choosing the direct predecessor of its backward partner.
- Vehicle_1 makes a gap with relation to vehicle 0, Vehicle_2 is making a gap with relation to Vehicle_1. The actual gap (in meters) depends on the actual speed (s) of the vehicles that are involved in the "gap-making.
- After finishing gap making / pairing the Platoon leader of A (Vehicle_1) goes into a merging status and hands over the Platoon lead flag to the next vehicle: Vehicle_3.
- Vehicle_3 starts pairing up with Vehicle_2 and creates a gap.
- When the gap is ready, vehicle from platoon B send out a message to its pairing partner (Vehicle_2 to Vehicle_1) to let it know it can start the merging sequence. The actual merging may be happening in parallel meaning Vehicle_3 may already start merging while Vehicle_1 is still in its merging status (finishing).
- The merging zone where the actual merging is happening is defined as: "as late as possible" so the vehicles use the available road before the lane closure.

  NB Information related to safety distances, timing calculations and safety measures will be announces in future check-list (if not already available in other deliverables).

Stage 4: Merging done / Platooning

- When the merging is done Vehicle_1 adapts the platoon ID of B.

The same sequence with the three stages will repeat for the other vehicles in platoon A and B. When all vehicles are merged the virtual RSU send out "EoS" (End of Scenario) message, meaning that the scenario and judging stops.

2.3 Intersection scenario

The figure below is based on the description of scenario 2 Intersection from [2].

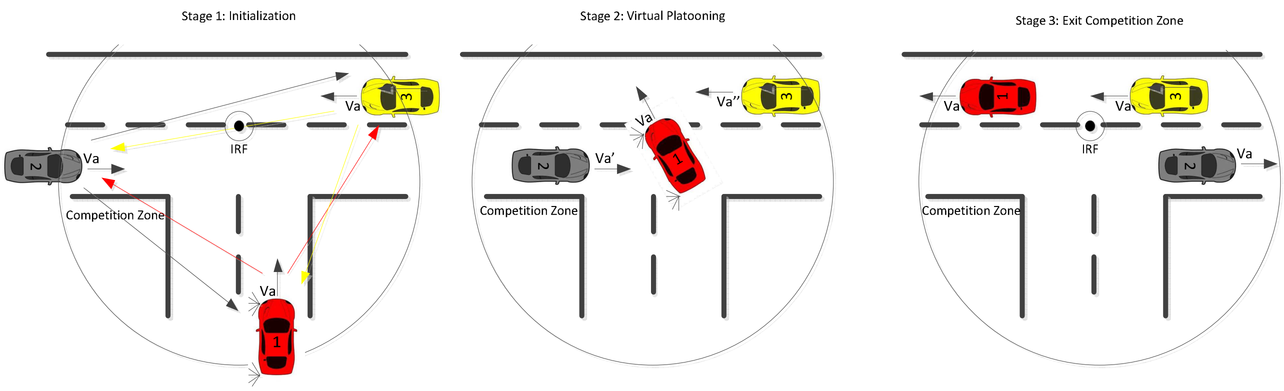

Figure 3 Intersection scenario

Starting points are:

- A RSU is being used for reproducibility purpose and management of the correct execution of the scenario, to make sure that the (pre)conditions are the same when (re)starting the scenario. This phase is called the preparation phase, it is not part of a judging criteria unless the required conditions at the start of the scenario are not met.
- The competing vehicles enter the Competition Zone (CZ) at (almost) the same time.
- The positions of the vehicles in the virtual platoon are fixed beforehand, based on priority lane number, to prevent any mismatch between the expected and actual entering time.
- The intersection data is known to each vehicle in advance (data file or digital map).
- From a safety point of view the contestants must implement "emergency brake". This means that when the algorithm for the virtual platoon stops for some (unknown) reason, the vehicle is expected to broadcast immediately an emergency brake message! Upon receiving this broadcast message the participating vehicles in the scenario must stop immediately.

Description of the scenario:

This scenario is based on creating a virtual platoon. Upon entering the CZ the vehicles start, based on a predefined order) decreasing their speed in order to form a platoon to each other as depicted in Figure 4.

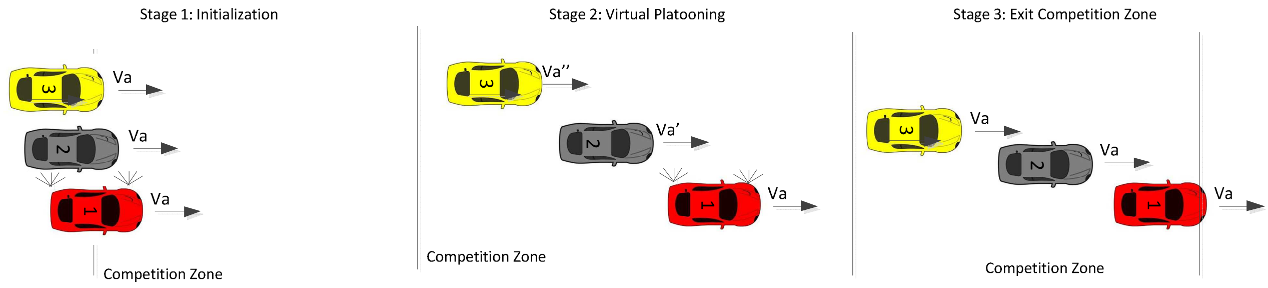

Figure 4 Virtual platooning

Stage 0: Preparation phase

Critical in this scenario is that the vehicles arrive at the CZ at almost the same time with the designated speed of 30km/h.

- Participating vehicles start in their standby / wait state upon receiving CAM message from RSU, meaning: switch on the controller, choose the lane, choose the scenario.
- Based on the trajectory this profile will be loaded in the controller to make sure that the arrival and speed will be aligned.
- The vehicles may calculate the ETAs (estimated time of arrivals) and broadcast these to the other competitors. This may become a judging criteria.
- The RSU sends out a "start" signal. Upon receiving this signal the participating vehicles start driving towards the designated speed of 30km/h in order to enter the CZ at almost the same time.

Stage 1: Initialization, entering the Competition Zone

The competing vehicles enter the CZ at almost the same time. Based on the intersection data each participating vehicle is calculating their own coordinates and orientation with respect to the Intersection Reference Point (IRP) and the travelled distance inside the CZ. This calculated information is broadcasted with other information like velocity, acceleration, their lane ID, intention and vehicle counter (this is the lane number on which it enters the CZ). Since the vehicles enter the CZ almost at the same time while they broadcast they also should receive the same broadcasted information from the other participants. Based on the calculated and received information and the defined vehicle counter, each vehicle starts calculating the desired or needed acceleration in order to form the (predefined) virtual platoon.

Stage 2: Intersection / Virtual platooning

In this stage the vehicles are forming the virtual platoon. Because of the predefined order Vehicle_2 knowns that it has to follow its virtual Platoon Leader: Vehicle_1. Based on the speed, acceleration and the intersection geometry it will decrease its speed to reach the required safe distance, making it possible for Vehicle_1 to cross the lane of Vehicle_2. Same principle for Vehicle_3, based on the information of Vehicle_2, it starts to decrease its speed until the save (virtual) distance from Vehicle_2 is reached, making it possible for Vehicle_1 to enter the lane of Vehicle_2 in a safe manner.

Stage 3: Exit Competition Zone

Upon exiting the CZ the vehicles should leave the virtual Platoon and if needed go to the normal platooning mode. Also on exit, the RSU will send a “EoS” message, meaning that the scenario and thus the judging ends.

2.4 Emergency vehicle scenario

Figure 5 is based on the description of scenario 3 Emergency Vehicle from [2]. This is a demonstrator scenario, meaning that is not a part of the GCDC competition, but the teams must be able to demonstrate such scenario. The scenario is not described in Deliverable 2.1, this section is a proposal to approach this scenario.

The emergency vehicle must be able to specify that it wants to drive either in one specific lane or in between two lanes. We use the information available in the current DENMs, without adding any extensions.

The lanes can be numbered as following (consistent with ETSI TS 102 892-2 [8]):

- Lane 1 is the outermost driving lane (in our case the right lane).
- Lane 2 is the second lane from outside (in our case the left lane).
- Lane 0 is the hard shoulder.

A basic assumption is that there is a higher probability that a road has a hard shoulder next to the outermost driving lane than next to the innermost driving lane. Thus it is easier for vehicles to move to the right than to the left. On this assumption we can create the following rules:

For vehicles in lane 1:

- If the emergency vehicle requested lane 0, move as far to the left as possible.
- If the emergency vehicle requested lane 1, move as far to the right as possible, either to the right part of lane 1, or, if hard shoulder is available, to that hard shoulder.
- If the emergency vehicle requested lane 2, move as far to the right as possible (same as if lane 1 was requested).

For vehicles in lane 2:

- If the emergency vehicle requested lane 0, move as far to the left as possible, either to the left part of lane 2, or, if hard shoulder is available on the left, move to that hard shoulder.
- If the emergency vehicle requested lane 1, move as far to the left as possible (same as if lane 0 was requested).
- If the emergency vehicle requested lane 2, move as far to the right as possible.

In general these rules tell: if the vehicle is in the requested lane or to the right of it, it should move right; if the vehicle is in the lane left of the requested lane, the vehicle should move left. This way, "one and a half" lane will be cleared for the emergency vehicle: the requested lane and a space between the requested lane and the next-towards-inside lane.

For this scenario we follow the original description meaning that the emergency vehicle will request only lane 1. (Lane 2 would also be possible but for simplicity lane 1 is sufficient.)

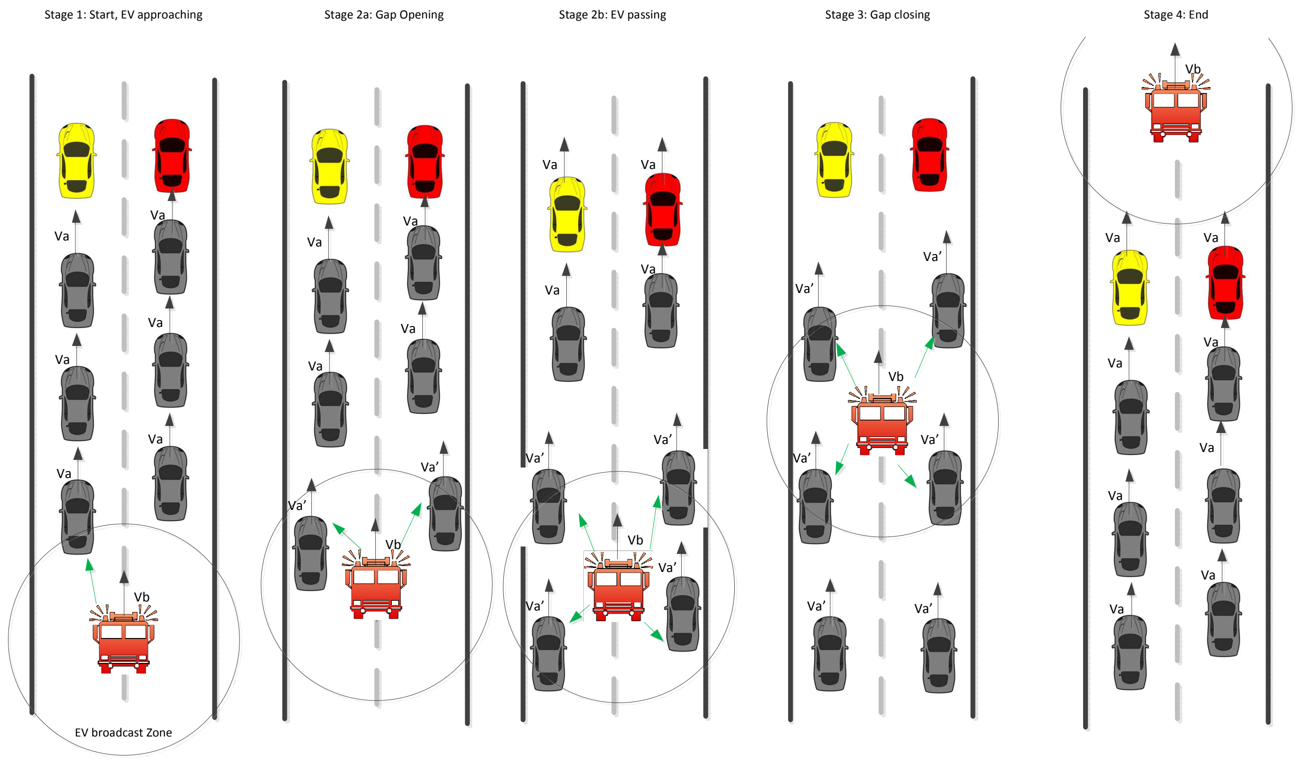

Figure 5 Emergency Vehicle demo scenario

Starting points:

- All vehicles have the same constant speed (50 km/h), forming 2 rows (left lane and right lane).
- The EV sends out the preferred lane it wants to pass on. On receiving this signal the vehicles know towards which side they should move in order to create room for the emergency vehicle.
- Because this scenario is a demo, the competitors will not be judged.

Stage 0: preparation

- Participating vehicles start in their standby / wait state upon receiving CAM message from RSU, meaning: switch on the controller, choose the lane, choose the scenario.
- RSU sends “start” message for both platoons start driving and accelerating towards a speed of 50km/h

Stage 1: Start, EV approaching

In this stage the two platoons are driving in a normal platooning way at a speed of 50km/h. An Emergency Vehicle (EV) is approaching at a speed of 80 km/h the tail of the two “platoons” (we use the word platoon, but the vehicles do not need to be platooning in this scenario).

Stage 2a: Gap opening

The EV has reached both platoons, in principle all vehicles are receiving the EV signal, but some may receive earlier. Upon receiving this signal the vehicle immediately stop synchronising its speed with the vehicles in front of them. The speed of the vehicles reduces to 40km/h and they are making room, by going to the sideways of the road. According to [4] all manoeuvring are executed manually except the reception of the EV approaching signal. It is up to the participants to decide if reducing the speed and making room is also automated or kept manual.

Stage 2b: Gap opening / EV passing

More vehicles in both lanes are receiving the EV broadcast signal (CAM, DENM or both), thus reducing their speed to 40km/h and going sideways.

Stage 3: Gap closing

The vehicles at the tail are not receiving the EV broadcast signal anymore. In this stage the vehicles not receiving the EV signal, will go back to the normal position on their lane. The vehicles at the tail also “see” vehicles in front of them and starting synchronising their speed and distance again.

Stage 4: End

In this stage all vehicles no longer receive the EV broadcast signal anymore. The vehicles are returning to their normal initial behaviour. The internal mechanism assures that the first vehicles accelerate to the normal speed of 50km/h. When all the vehicles are back at the initial speed an EoS message will be send in order to notify the contestants that the scenario has ended.

3 Interactions flow in the scenarios

The interaction protocols [2] that are needed in the scenarios will be investigated based on the identified stages. The stages are being used to determine the message flow diagrams during each stage. Out of the flows the information that needs to be exchanged can be determined and what the associated data types and characteristics are.

3.1 Merging interactions

3.1.1. Message Flow Diagrams

To perform the merging scenario presented in the above chapter, messages are designed for the purpose of forward pairing, backward pairing, safe to merge, as well as the platoon leader flag handover. We take the simplified merge scenario shown in Figure 6 where two vehicles a_{i-1} and a_i from platoon A merge with platoon B as an example, and show the information flow for the whole merge process in Figure 7.

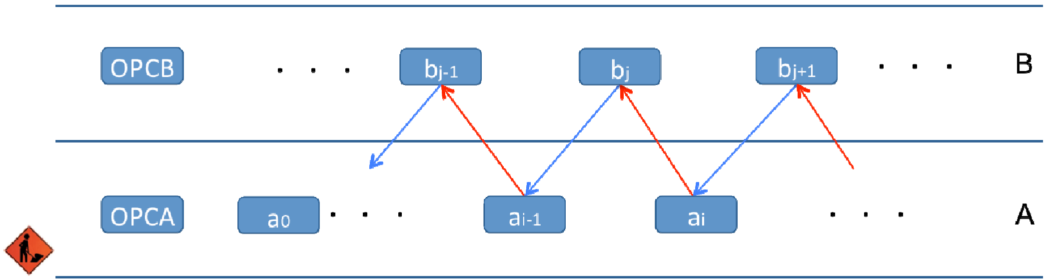

Figure 6 Illustration of the cooperative platoon merge scenario

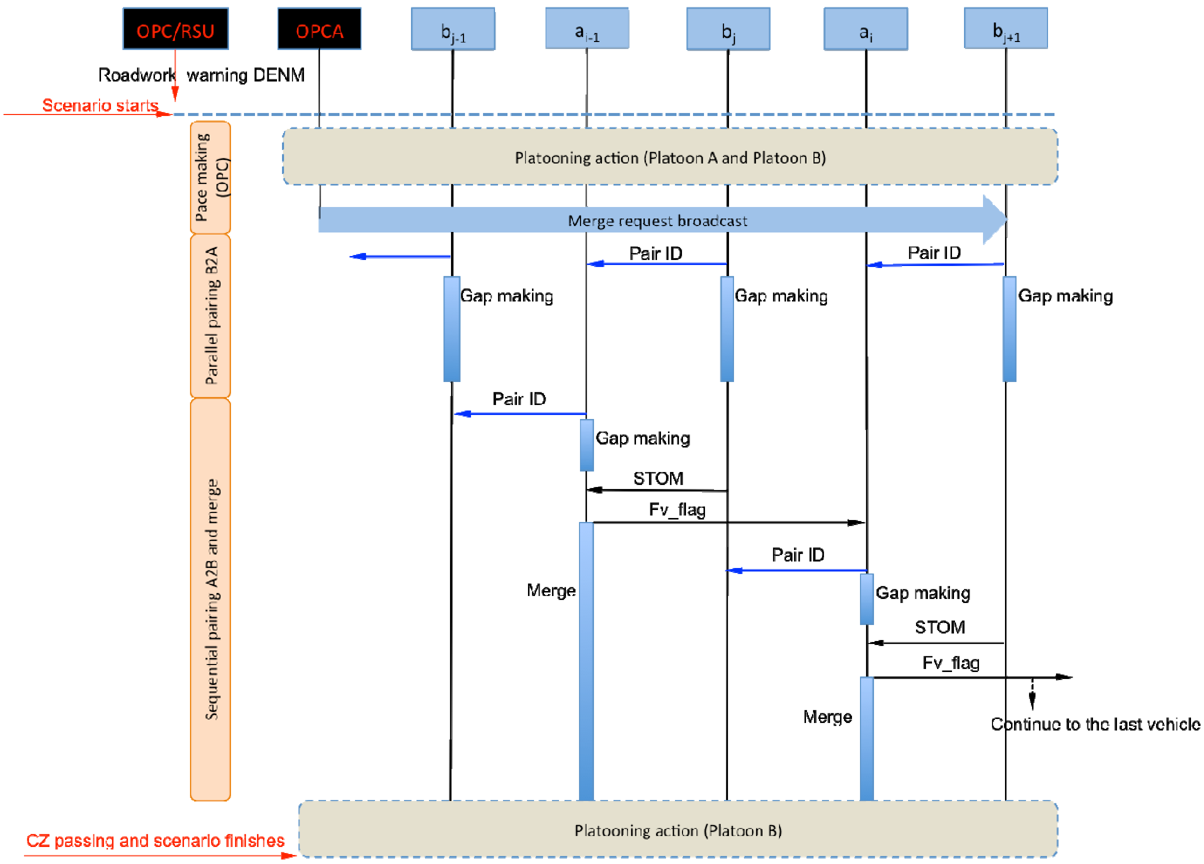

Figure 7 Message flow for platoon merge

As illustrated in Figure 7 and also described in detail in Deliverable D2.1 the interaction protocol, the interaction and message exchanges include several stages, namely pace making, B2A pairing, A2B pairing, FV (First Vehicle) handover and merge, platoon action. We discuss in detail the information exchange needed for each of the stages.

- **Pace making**: Before competition starts, vehicles at both of the lanes are running in platoon mode but with different speeds. Platoon A is running with a speed of 80 km/h while platoon B is running with lower speed of 60 km/h. One of the OPC vehicles emulates a RSU and sends out "*roadwork warning"* messages, starting the competition and the pace making process. Vehicles within platoon A start to reduce their speed and synchronize with those within platoon B. When OPC vehicles at both of the lanes are on the correct position and vehicles within the two platoons have their speed synchronized, OPC vehicles at lane A, i.e. OPC_A, broadcasts out a request for merge message and starts the merge process. Three major messages are involved i.e. *roadwork warning DENM*, *platooning information* and *request for merge*:
    - *Roadwork warning DENM*: The message is a periodically broadcast message that is triggered because of road works. It notifies all vehicles within a certain area about the roadwork, together with detailed information including the roadwork location, the speed limitation, the size of the roadwork zone, etc. At GCDC, this message is sent out from one of the OPC vehicles.

      Since the whole scenario is triggered by road works, another function for *roadwork warning DENM* is to start the challenge. Upon receiving the first *roadwork warning DENM*, the competition starts and thus the pace making process.
    - *Platooning information*: During the pace making period, vehicles in platoon A and platoon B share their dynamics within their platoon members. The information, broadcast periodically by each of the platoon members, is used to maintain normal platooning operation i.e. cooperative adaptive cruise control (CACC).
    - *Merge request:* Once OPC vehicles are in position and vehicles at the two lanes have their speeds synchronized, OPC_A broadcasts *merge request* to start the merge process.
- **B2A pairing**: The reception of *merge request* starts the merge process with B2A pairing as the first step. This stage of B2A pairing involves paring of vehicles from lane B with vehicles at lane A. This is a stage with parallel activities, i.e. vehicles in platoon B broadcast information in parallel to vehicles in platoon A. The information involved in this phase is *Pair ID* that denotes for each of the vehicles in platoon B the targeting vehicle at platoon A for pairing:
    - *Pair ID*: When B2A pairing starts, vehicles b_{j-1}, b_j, b_{j+1} at lane B identify their targets simultaneously. For each of the vehicle at lane B, its target is the closest front vehicle on lane A, i.e., $b_j \rightarrow a_{i-1}$, $b_{j+1} \rightarrow a_i$. The target is called forward most-important-object on left lane (FWD MIOL) and it will be the pairing partner for the vehicle at lane B. Once pairing partners are identified, vehicles at lane B set the Pair ID and broadcast this information.

  In an optimal setting, each vehicle in lane B will have a pair in lane A that is called forward pair (FWD pair), while the vehicle itself is called backward pair (BWD pair) for its FWD pair. For example, vehicle a_{i-1} is the FWD pair for vehicle b_j while b_j is the BWD pair of a_{i-1}. Notice that a vehicle in lane B may have no pairing vehicles at lane A, thus continuing to run in CACC mode.
- **A2B Pairing**: This stage involves a series of consecutive pairing processes where vehicles at lane A pair with vehicles at lane B one after one. Compared to B2A pairing, the pairing processes for A2B is active only for the FV in platoon A. For example, assuming a_{i-1} is the current FV at platoon A. After A2B paring, b_j pairs with a_{i-1}, thus b_j is the BWD pair of a_{i-1}. For the period of A2B pairing, a_{i-1} sets its pairing target at lane B as its BWD pair's predecessor, i.e. b_{j-1}, thus pairing with it. When A2B pairing finishes, e.g. when a_{i-1} pairs with b_{j-1}, and b_j opens enough for a_{i-1} to merge, a *safe-to-merge message*

(STOM) will be broadcast out from b_j and triggers next stage. This stage involves mainly two sets of information, *Pair ID* and *safe-to-merge (STOM)*:

  - *Pair ID*: Each time, FV at lane A set its FWD pair (*Pair ID*) as the predecessor of its BWD pair for A2B forward pairing and broadcast the information out through V2V.
  - *STOM*: *STOM* is used for the current FV's BWD pair to confirm that the gaps are ready and merge can start. For the current FV at lane A, say a_{i-1}, once gaps are ready for merge, i.e. a_{i-1} opens enough gaps and b_j maintains enough gaps to a_{i-1}, b_j confirms the status and sends out *STOM* to inform FV a_{i-1} to merge.

- **FV handover and Merge**: This stage involves FV handover process that changes the FV status to next vehicle within the platoon, and the merge execution. Upon receiving *STOM*, FV at lane A first transfers the FV status to its successor and then enters the status of merge. Once the FV status is changed, the new FV starts a new round of A2B pairing and merge process. Notice that the new FV does not need to wait until the previous FV to finish the merge. This stage makes sure that there is only one FV at the platoon and only FV can start the A2B pairing process. This stage involves one information exchange, the *Fv_flag*:
  - *Fv_flag*: This flag simply indicates that the vehicle status i.e. whether the vehicle is a FV or not. Once the current FV changes its status from FV to non-FV, its successor detects and sets its status to FV, thus starting a new round of A2B pairing and merges.
- **Platooning action**: This stage involves the actual platooning action. Vehicles in platoon B keep CACC during the whole process. Once FV in platoon A leaves platoon A and merges with platoon B, it will be a member of the platoon B, thus entering the platoon mode. Similar to the situation at the beginning of the scenario, *platooning information* is needed during this stage to maintain the new platoon B.

3.2 Intersection interactions

3.1.2. Message Flow Diagrams

As illustrated in Section 2.3 and also in Deliverable D2.1 the scenario of cooperative intersection, three vehicles approach a non-signalized T-shape intersection and need to coordinate the intersection passing safely and efficiently. For showing the information flow, we abstract the scenario in

Figure 8, where vehicles 1, 2, and 3 need to coordinate their intersection passing when entering the competition zone (CZ).

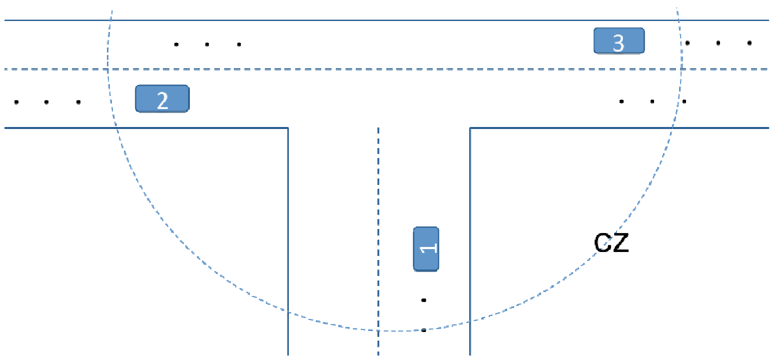

Figure 8 Illustration of the cooperative intersection scenario

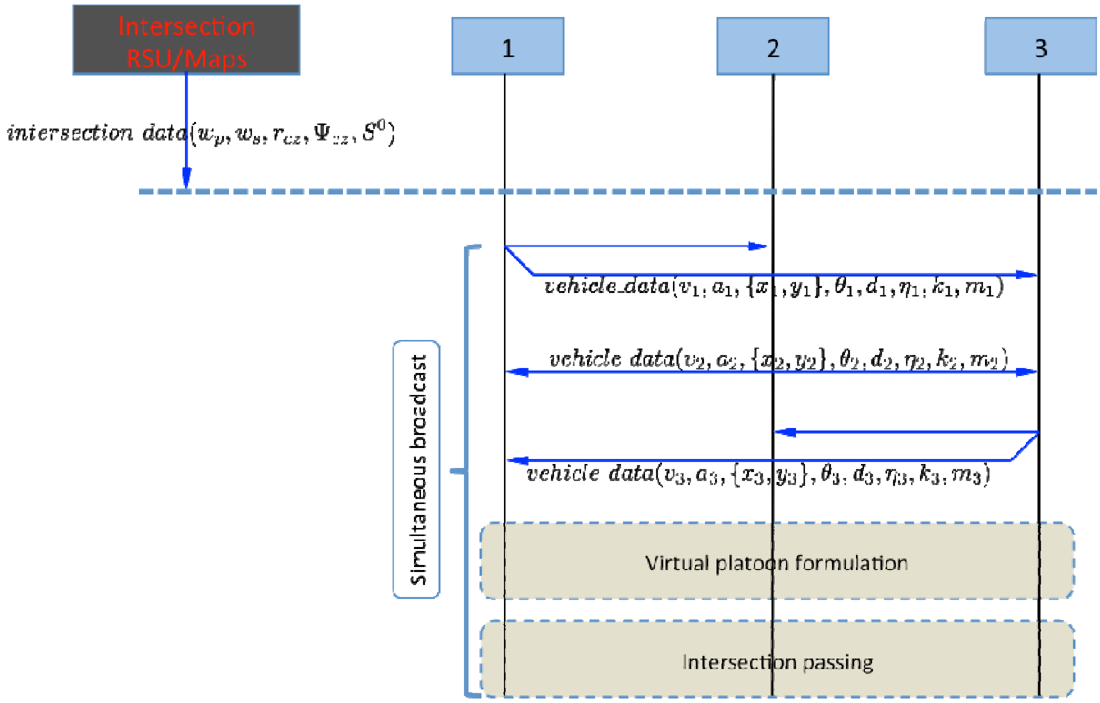

Figure 9 Message flow for intersection coordination

Figure 9 illustrates the signal exchanges during the intersection coordination. Detailed intersection data will be provided on forehand and must be integrated with the vehicles digital maps. A concept of virtual platoon is introduced, where the three competing vehicles formulate a virtual platoon to pass the intersection. To formulate the virtual platoon, upon entering the competition zone, the vehicles need to share certain information with each other to exchange the order of the virtual platoon. Then the controllers will control the vehicles to pass the intersection. Generally speaking, there are three stages involved, initialization, virtual platooning, and intersection passing. Mainly two sets of messages, the *intersection data* and the *vehicle data* messages of each of the vehicles, are involved. And only *vehicle data* needs to be exchanged in real time.

- **Initialization**: At this stage, vehicle prepare to enter the competition zone. As specified in the scenario, vehicles will arrive at the predefined location with predefined velocity and almost at the same time. No information exchanges are needed at this stage.
- **Virtual platooning**: This is the core stage, as already mentioned in Section 2.3, meaning that the participating vehicles coordinates their trajectories for passing the intersection safely and efficiently. Upon entering the CZ, vehicles start to broadcast information of their own status and receiving information from others. The following information flows can be distinguished:
    - *Intersection data*: *Intersection data* contains static geometry information including CZ radius, road width, road priority, angle between the primary and secondary roads, and so on. To facilitate the challenge, it is provided beforehand as offline data, this can be just a text file or e.g. as digital map. Detailed specifications on the message contents are described in the next chapters on message definition and can also be found in D2.1.
    - *Vehicle data*: *Vehicle data* messages include relevant information that will be used by the vehicle controllers to formulate a virtual platoon. Those messages form the major information to be exchanged during the intersection passing. This is specified in the next chapter and is also described in D2.1, the information includes vehicle velocity, acceleration, vehicle coordinates, vehicle orientation, driving intention, the running lane, travel distances within the CZ, as well as a counter indicating the number of vehicles within the intersection.

- **Intersection passing:** After the virtual platoon is formed, the vehicles will be controlled by platoon controllers, thus executing the intersection passing. *Vehicle data* will be periodically broadcast throughout the scenario.

3.3 Emergency Vehicle interactions

3.1.3. Message Flow Diagrams

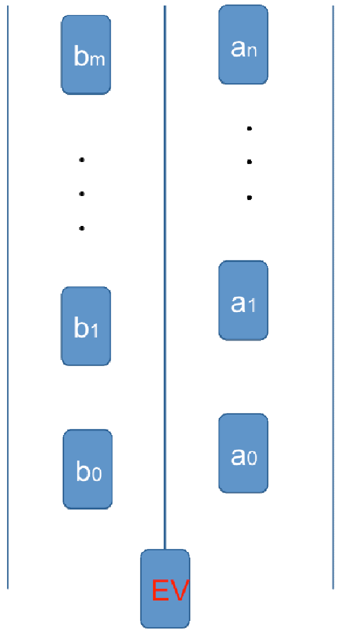

Figure 10 Illustration of the emergency vehicle scenario

As described in Deliverable D1.1 and the previous section, the EV scenario involves an EV and a number of other cooperative vehicles. As illustrated in Figure 10, an EV is approaching a two-lane road with both of the roads with traffic. Vehicles on both of the lanes, e.g., a_0, a_1, …, a_n, and b_0, b_1, …, b_m, must give way to the EV upon receiving information on how EV intends to pass by.

Figure 11 illustrates the information flow for the EV scenario. The scenario can be divided into phases including EV approaching, gap opening, EV passing, and gap closing. We discuss for each of the phases the information needed.

- **EV approaching**: When an EV is on the road, it broadcasts messages about its status through CAM messages. Furthermore, when EV needs a free path and there is traffic on the road, EV broadcasts DENMs that notify the vehicles how EV wants to pass through, at what lane. There are two sets of messages involved in this phase, standardized CAM and DENM:
  - CAM: In addition to the data fields such as vehicle dynamics, CAM includes the status of the EV, e.g. notifying that this is an EV.
  - DENM: This DENM is triggered by the EV when EV wants a free lane on the road. Compared to CAMs, DENMs have more detailed information of the EV such as how the EV wants to travel through the road i.e. from which lane, the direction, how large is the relevant area etc.

  Notice that both CAMs and DENMs are standardized. Detailed specification on the data fields can be found in the rest of this deliverable and also in related ETSI standards [5],[6].

- **Gap opening**: When the participating vehicles receives the CAMs and DENMs from the EV, the vehicles understand how the EV wants to pass through, thus making gaps to give way to the EV. It is natural that vehicles at the tail such as a_0 and b_0 receives this message first. It is also possible that a_1

and b_1 receive the DENM at the same time with a_0. In any case, when vehicles receive DENMs, they evaluate the relevance and take actions accordingly. At this stage, the same messages (CAMs and DENMs) as in the previous phase are needed.

- **EV passing**: In a perfect scenario, EV broadcast CAMs and DENMs well ahead of time, and when EV approaches tails of each other vehicle, gaps should be already open and EV can pass through with no stop or even any deceleration. During the whole passing stage, EV broadcasts CAMs and DENMs.
- **Gap closing**: Once EV has passed, vehicles may return to their normal driving lane. This is triggered when the reception of DENMs is out of reach or negation of DENMs is received. When DENMs are out of reach or vehicles are out of the relevance area, vehicles assume that the traffic returns to normal and they will return to their original lane. It is also possible that EV sends out a negation DENM that indicates that EV approaching DENM has expired for vehicles it has passed, thus cancelling the EV approaching event. In any cases, vehicles have the ability to check the relevance of DENMs and make decisions accordingly, for example by check the EV position.

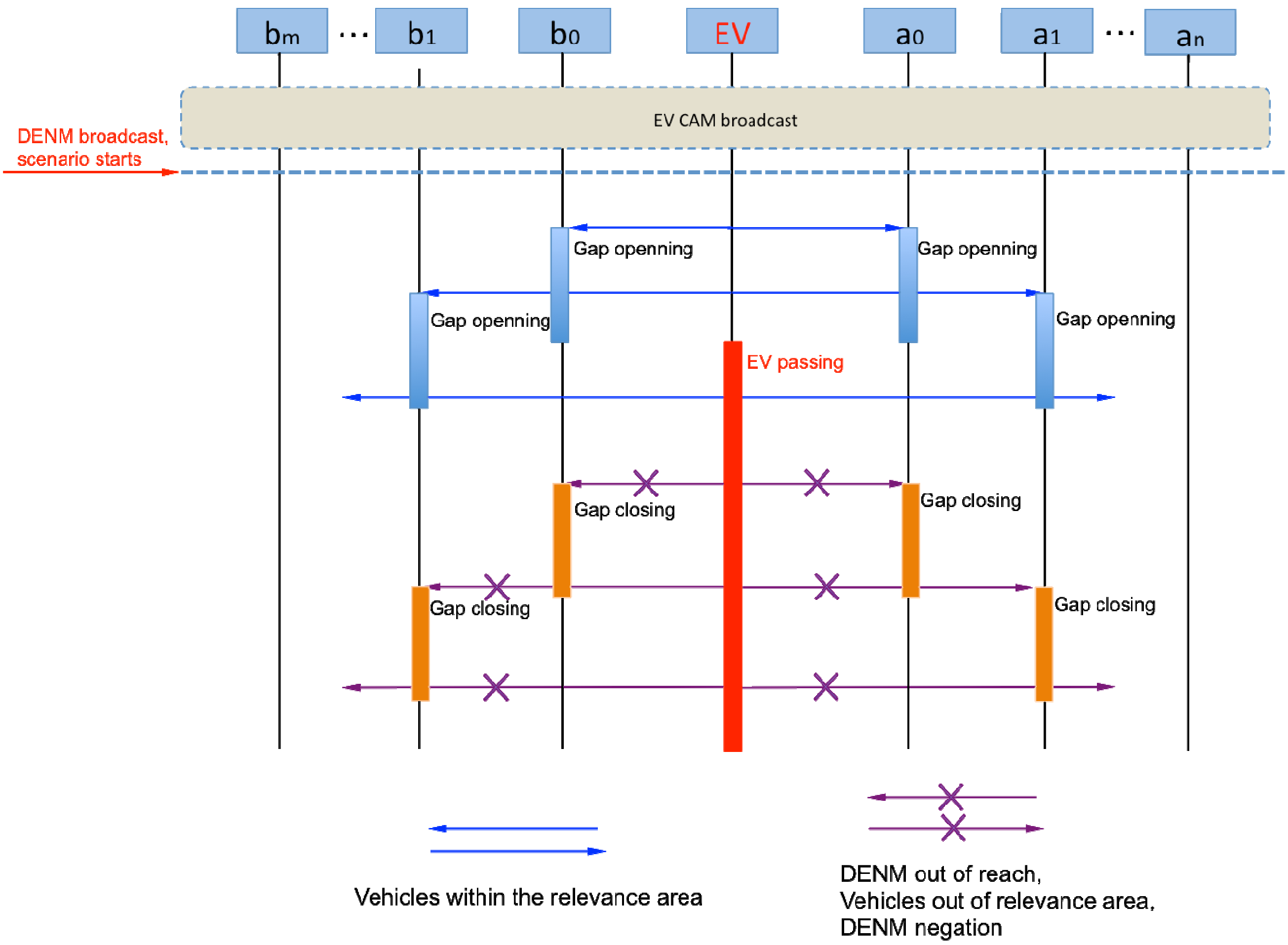

Figure 11 Message flow for the emergency vehicle scenario

4 Message sets for interaction

4.1 Introduction

The efficient realisation and the correctness of the interaction protocols is depending on the reliability of the communication protocol. In the i-GAME project, it is required that all the participating teams use the existing communication architecture as defined in i-GAME D3.1 [1]. In this chapter, we focus on the complete message set that is needed for the proposed interaction protocols of all scenarios described in the deliverable D2.1 [2]. For the GCDC2016, the interaction protocols involve intensive communication between vehicles. The message sets that are defined in C-ITS such as CAM and the DENM are not enough to support all control actions. An additional message set is defined for specific control purposes, and also some adaptions of the current message sets are needed. The definition of additional message sets are done in such a way that, when it is applicable, the messages can be easily adopted in current available message sets. All the work within this project, coming from a bottom-up practical approach does have the potential to contribute to the current ETSI ITS standard, especially for cooperative safety applications.

Besides the current ETSI standards there are also similarities between the i-GAME work and the Autonet2030 project. Autonet2030 also defines additional message set and extensions [7]. However the difference is that Autonet2030 aims for a future platooning application, while the i-GAME goal is to develop real-life implementations, from earlier experience with GCDC2011 and using existing equipment, which can be tested and implemented by all the teams in GCDC 2016. The i-GAME project strives as much as possible for synergy between projects and may adopt some initial message set that already is designed in Autonet2030.

The necessary message set for performing the i-GAME scenarios is based on the set of signals required by in-vehicle control layer and the information flows as described in Chapter 3. In the following paragraphs the complete list of needed messages will be divided into two main categories: messages that are available in the ETSI standard, i.e., CAM, DENM and messages that does not exist, need to be created in a i-GAME message set.

4.2 Required Messages

In cooperative ITS, communication between vehicles and between vehicle and RSU play a crucial role. The information is sent in the form of different messages depending on the purpose. For i-GAME interaction protocols, we have defined a set of important signals that are needed to exchange between vehicles or between vehicles and a RSU. All the proposed interaction protocols have been tested via simulation and/or in real-life implementations. So all signals needed for the messages have been verified, see i-GAME D2.1 [2]. The goal in this document is to map the required signals into the existing standard messages like CAM and DENM or into a new i-GAME messages.

As illustrated in the table 1, there are more than 40 messages needed to be exchanged during the execution of the scenarios. These messages can be classified into the following categories:

- State: including vehicle related characteristics, velocity and position.
- Control: including parameters used for control purpose, i.e. measured distance from a vehicle to a near object (safety distance)
- Event-driven: including information about some events happen, such as road works or an emergency vehicle approaching

Due to very strict safety measures there is an important deviation for the CAM and DENM standard implementation at the facility layer. The facility layer normally broadcasts at a maximum of 10Hz frequency rate. For the CAM and i-GAME messages it is important that these are transmitted at a higher frequency, 25Hz. This is primarily needed from safety point of view to reduce delays that could lead to reduced safety distances. We want to have a safe scenario execution for the GCDC 2016. And in addition this enables practical scenario execution without extreme vehicle gap openings or slow manoeuvring which would deviate from real life vehicle traffic behaviour. The complete list with the messages, the frequency and the protocol set is displayed in Table 1 – V2V message set. These messages are the minimum needed to execute the scenarios. See Appendix A for detailed description of the required data elements within these messages.

Table 1 – V2V message set

ID	Message description	Protocol	Frequency rate
1	Header	CAM/DENM/iCLCM	-
2	GenerationDeltaTime	CAM	25Hz
3	Station ID	CAM/DENM/iCLCM	25Hz
4	Station Type	CAM	25Hz
5	Vehicle Role	CAM	1Hz
6	Vehicle length	CAM	25Hz
7	Vehicle rear axle location	iCLCM	25Hz
8	Vehicle width	CAM	25Hz
9	Controller type	iCLCM	25Hz
10	Vehicle response time constant	iCLCM	25Hz
11	Vehicle response time delay	iCLCM	25Hz
12	ReferencePosition (latitude, longitude, confidence)	CAM	25Hz
13	Heading (Heading, confidence)	CAM	25Hz
14	Speed	CAM	25Hz
15	YawRate	CAM	25Hz
16	Longitudinal vehicle acceleration	CAM	25Hz
17	Desired longitudinal vehicle acceleration	iCLCM	25Hz
18	MIO ID (measured by object vehicle)	iCLCM	25Hz
19	MIO range (measured by object vehicle)	iCLCM	25Hz
20	MIO bearing (measured by object vehicle)	iCLCM	25Hz
21	MIO range rate (measured by object vehicle)	iCLCM	25Hz
22	Time headway	iCLCM	25Hz
23	Cruise speed	iCLCM	25Hz
24	Merge request flag	iCLCM	1Hz
25	Safe-to-merge (STOM) flag	iCLCM	1Hz
26	Merging flag	iCLCM	1Hz
27	ID of fwd pair partner	iCLCM	1Hz
28	ID of bwd pair partner	iCLCM	1Hz
29	Tail vehicle flag	iCLCM	1Hz
30	Head vehicle flag	iCLCM	1Hz
31	Platoon ID	iCLCM	1Hz
32	Travelled distance inside the CZ	iCLCM	25Hz
33	Intention (left, right, or straight)	iCLCM	1Hz
34	Lane on which the vehicle enters the CZ	iCLCM	1Hz
35	Intersection vehicle counter	iCLCM	1Hz
36	Pair acknowledge flag	iCLCM	1Hz
37	Reference Time	DENM	10Hz

ID	Message description	Protocol	Frequency rate
		(management container)	
38	EventType (Roadworks, Stationary vehicle, Emergency vehicle approaching, Dangerous Situation (Emergency electronic brake light)	DENM (situation container)	10Hz
39	ClosedLanes	DENM (alacarte container	10Hz
40	LanePosition	DENM (alacarte container)	10Hz
41	Participants ready	iCLCM	1 Hz
42	Start scenario	iCLCM	1 Hz
43	EoS (End of Scenario)	iCLCM	1Hz
44	Reserve/spare/future use	iCLCM	-
45	Reserve/spare/future use	iCLCM	-
46	Reserve/spare/future use	iCLCM	-

The messages as described in Appendix A are also classified according to the scenarios that use the messages. The information is available in the table of Appendix B.

4.2.1 Review of existing and related message set: CAMs, DENMs and Autonet2030

Both CAM and DENM messages are generated at the facility layer and do not depend on the application use cases. As mentioned earlier, we want to use the current message set as defined in the ETSI standard as much as possible, therefore, we review the existing CAM and DENM definitions before we introduce an i-GAME message set.

Cooperative Awareness Messages (CAMs)

Cooperative Awareness Messages (CAMs) are messages exchanged in the ITS network between ITS-S to create and maintain awareness and to support cooperative performance of vehicles using the road network. A CAM contains status and attribute information of the originating ITS-S. The content varies depending on the type of the ITS-S. For vehicle ITS-S the status information includes time, position, motion state, activated systems, etc. and the attribute information includes data about the dimensions, vehicle type and role in the road traffic, etc. On reception of a CAM the receiving ITS-S becomes aware of the presence, type, and status of the originating ITS-S. The received information can be used by the receiving ITS-S to support several ITS applications.

CAM messages contain information about vehicle position, driving dynamic, physical attributes, etc. CAMs are broadcasted typically with a frequency of 1Hz to 10Hz. The packet size of CAM varies from 326 byte (update rate from 1-10Hz) to 402 byte (update rate of 1-2Hz). Without the header part, the CAM useful data is only 14 to 90 bytes long. We do not mandate the use of a security header and signature.

Decentralized Environmental Notification Message (DENM)

This message is mainly used by ITS applications in order to alert road users of a detected event using ITS communication technologies. DENM are event-based generated, i.e., intersection collision risk warning or longitudinal collision risk warning and consist of information related to such an event. Similar to CAMs, DENM will also be transmitted periodically, with a certain update rate (typically 1-20 Hz) until a timer expires.

Autonet2030 message set

These messages had been created on multiple scenarios, each with a message format specification. For i-GAME the relevant scenarios are:

- CLCM (Cooperative Lane Change Message): This handles the interactions between ITS stations in order to plan, prepare and execute a cooperative lane change.
- CMM (Convoy Management Message): This allows cooperative vehicles to plan automated adjustments of their speed and heading according to a decentralized mechanism, or to plan ahead needed lane change manoeuvres.
- IEM (Intersection Entry Message): This is a service that supports the communication between vehicle and intersection controller for priority based coordination of autonomous vehicles at intersections.
- CSM (Cooperative Sensing Message): This is a facilities-layer component for sharing sensed data with neighbouring cooperative ITS Stations.
- CSAM (Cooperative Speed Advising Message): This is to reliably deliver the advised driving speed for a given road segment, through a scalable broadcast-based approach.

4.2.2 Message set extensions / proposal of new message set

For platoon merging scenario, it is not enough to use only original CAM and DENM message set. CAM has no data fields that can be used for pairing signal, MIO information (e.g. ID 18). DENM is a good candidate for the STOM message (ID 25), however, the available place in the DENM data field, that can be used, is very limited in its value options. In addition, there are several reasons to propose a new message set for i-GAME. Both CAM and DENM have varying update frequency depending on the vehicle dynamicity, which makes it difficult to control platooning. It is not cost efficient to modify current CAM and DENM because of several OBU (framework) implementations already have working decoders/encoders for the existing CAM/DENM standards. From a practical point of view it will be a lot easier for participants to just add the i-GAME specific messages.

Thus our i-Game message set will include current **CAM**, **DENM** and a new **i-GAME Cooperative Lane Change Message (iCLCM)** is defined, see table below.

Table 2 i-GAME Cooperative Lane Change Message (iCLCM)

ID	Protocol (ASN.1)
1	iCLCM.header.messageID.iclc
3	iCLCM.header.stationID
7	iCLCM.iclcm.VehicleContainerHighFrequency.vehicleRearAxleLocation
9	iCLCM.iclcm.VehicleContainerHighFrequency.controllerType
10	iCLCM.iclcm.VehicleContainerHighFrequency.vehicleResponseTimeConstant
11	iCLCM.iclcm.VehicleContainerHighFrequency.vehicleResponseTimeDelay
17	iCLCM.iclcm.VehicleContainerHighFrequency.targetLongitudinalAcceleration
18	iCLCM.iclcm.MIO.ID
19	iCLCM.iclcm.MIO.Range
20	iCLCM.iclcm.MIO.Bearing
21	iCLCM.iclcm.MIO.RangeRate
22	iCLCM.iclcm.VehicleContainerHighFrequency.timeHeadway
23	iCLCM.iclcm.VehicleContainerHighFrequency.cruiseSpeed
24	iCLCM.iclcm.Merge.mergeReq

ID	Protocol (ASN.1)
25	iCLCM.iclcm.Merge.STOM
26	iCLCM.iclcm.Merge.Flag
27	iCLCM.iclcm.PairID.FwdID
28	iCLCM.iclcm.PairID.BwdID
29	iCLCM.iclcm.Merge.flagTail
30	iCLCM.iclcm.Merge.flagHead
31	iCLCM.iclcm.S.PlatoonID
32	iCLCM.iclcm.S.DistanceTraveledCZ
33	iCLCM.iclcm.S.Intention
34	iCLCM.iclcm.Lane
35	iCLCM.iclcm.S.Counter
36	iCLCM.iclcm.PairID.AckFlag
41	iCLCM.iclcm.LowFrequencyContainer.ParticipantsReady
42	iCLCM.iclcm.LowFrequencyContainer.StartPlatoon
43	iCLCM.iclcm.LowFrequencyContainer.EoS

4.2.3 Message set implementation

The iCLCM message table from the previous section is used in the next chapter for message packet format specification and ASN.1 definitions; detailed information on that is available in the Appendices.

With our message set we want to speed-up real life cooperative automated driving implementations (with a focus on cooperative safety applications) by also contributing to standardization efforts. Obviously, our implementation is bounded by the defined scenarios and interaction protocols. We used a bottom-up and practical approach, where we build upon past experience, proven solutions and the available technology and equipment of today. This approach should work towards a smooth, controlled and safe execution of the scenarios by the participants.

As a result of this approach it is decided to use CAM and DENM and add iCLCM messages for the GCDC scenarios. At the facility layer we use non-standard frequencies of 1 up to 25Hz for CAM and also for iCLCM. This high frequency for CAM is needed due to very strict safety measures. CAM is originally not designed to support this automated driving and safety applications, so the CAM information and the iCLCM that we need for execution of the scenarios are mostly needed at this higher frequency. For the correct execution of the scenarios the support of 25Hz for CAM and iCLCM is mandatory since this reduces the delays that could lead to reduced safety distances. And this also enables practical scenario execution without extreme vehicle gap openings or slow maneuvering which would deviate from real-life vehicle traffic behavior.

We do not expect that these higher frequencies will be adapted for CAM in future facility layer standards. It is to validate that new message sets like the iCLCM are needed operating at higher frequencies are needed in order to facilitate cooperative automated driving and safety applications. The proposed iCLCM implementation is now designed as a 'CAM-like' message, based mainly on practical reasons. For future developments it can be more logical to use an event-based type message, but still this requires higher update rates than nowadays used in typical applications.

5 Message Packet format specification

5.1 CAM

A CAM is composed of one common ITS PDU header and multiple containers, which constitute a payload. The ITS PDU header is a common header that includes the information of the protocol version, the message type and the ITS-S ID of the originating ITS-S.

For vehicle ITS-S a CAM shall comprise one basic container and one high frequency container, and may also include one low frequency container and one or more other special containers:

- The basic container includes basic information related to the originating ITS-S.
- The high frequency container contains highly dynamic information of the originating ITS-S.
- The low frequency container contains static and not highly dynamic information of the originating ITS-S.

The special vehicle container contains information specific to the vehicle role of the originating vehicle ITS-S., this container is not needed within i-GAME. All CAMs generated by a RSU ITS-S shall include a basic container and optionally more containers. In Figure 12 the i-GAME messages are mapped that fit within the CAM definition.

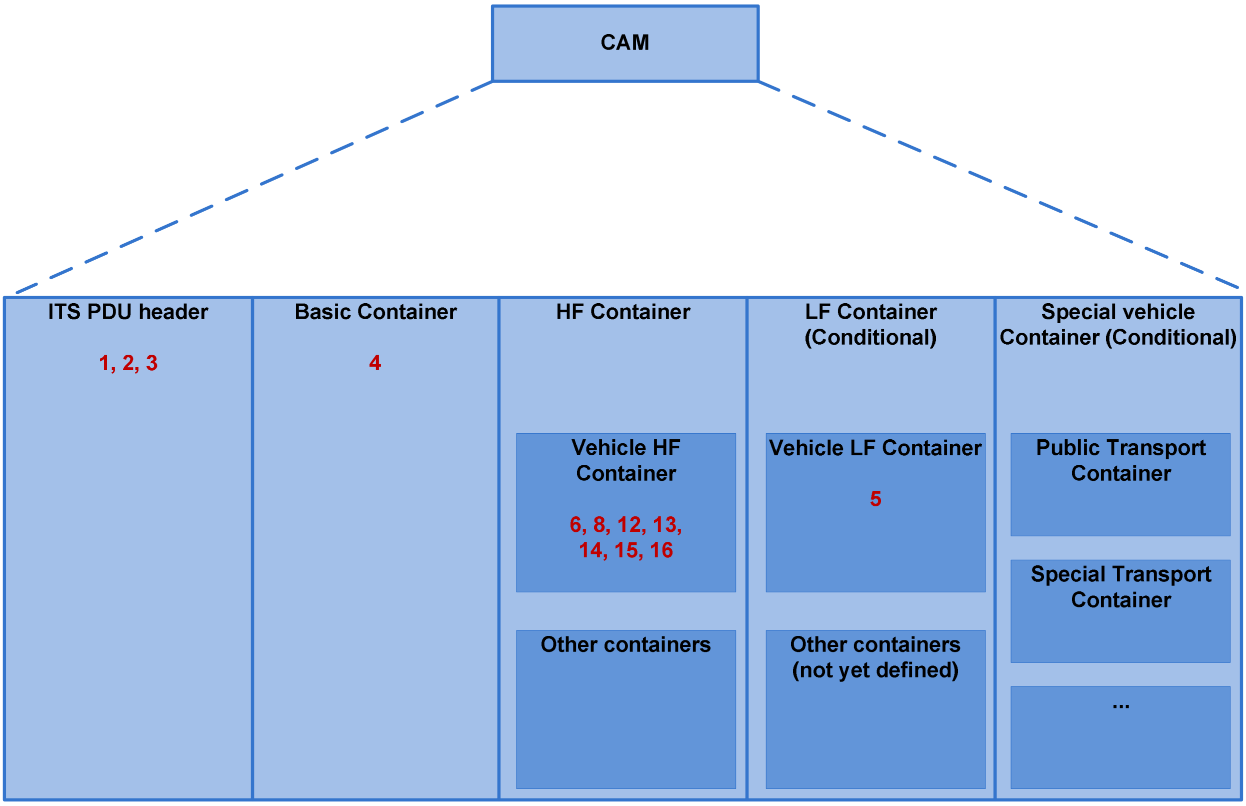

Figure 12 General structure of CAM format

5.2 DENM

A DENM is composed of a common ITS PDU header and multiple containers, which constitute the DENM payload. The ITS PDU header is common header that includes the information of the protocol version, the message type and the ITS-S ID of the originating ITS-S.

The DENM payload consists of 4 fixed order parts: the management container, the situation container, the location container and the à la carte container:

- The management container contains information related to the DENM management and the DENM protocol.
- The situation container contains information related to the type of the detected event.
- The location container contains information of the event location, and the location referencing.
- The à la carte container contains information specific to use case which requires the transmission of the additional information that is not included in the three previous containers

For all types of DENM, the ITS PDU header and the management container shall always be present. The situation container, the location container and the à la carte container are optional containers. For a cancellation DENM or a negation DENM, the situation container, location container and à la carte container shall not be present, the location container shall be present as well. The à la carte container is present only when applicable as specified in application specification standards [18], [19]and [21]. Figure 13 shows the mapping of the i-GAME messages on the DENM format.

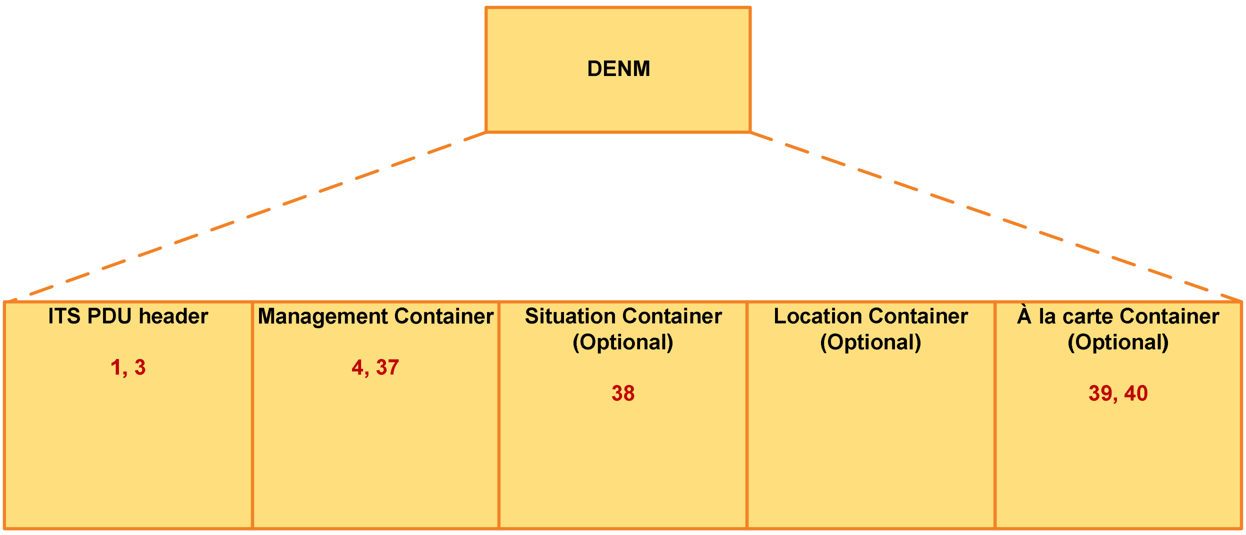

Figure 13 General structure of DENM format

5.3 iCLCM

An i-GAME Cooperative Lane Change Message (iCLCM) is composed of a common ITS PDU Header and multiple containers, which constitute the iCLC payload. The ITS PDU header is a common header that includes the information of the protocol version, the message type and the ITS-S ID of the originating ITS-S.

The iCLCM payload consists of 7 fixed order parts:

- The ITS PDU Header: containing the message ID, station ID.
- The iCLCM header with the generation time.
- The Vehicle HF container, containing additional information about the vehicle on a high frequency rate.
- The LF container, containing information that does not need to be updated at high frequency
- The MIO, this container holds information about the neighbours, the Most Important Objects.

- The Lane, the lane ID on which the vehicle enters the CZ.
- The Pair ID, containing the station ID of the pairing partner.
- The Merge container, all information needed to perform the actual merging.
- The S container: a container for additional information that is needed for the scenario execution.

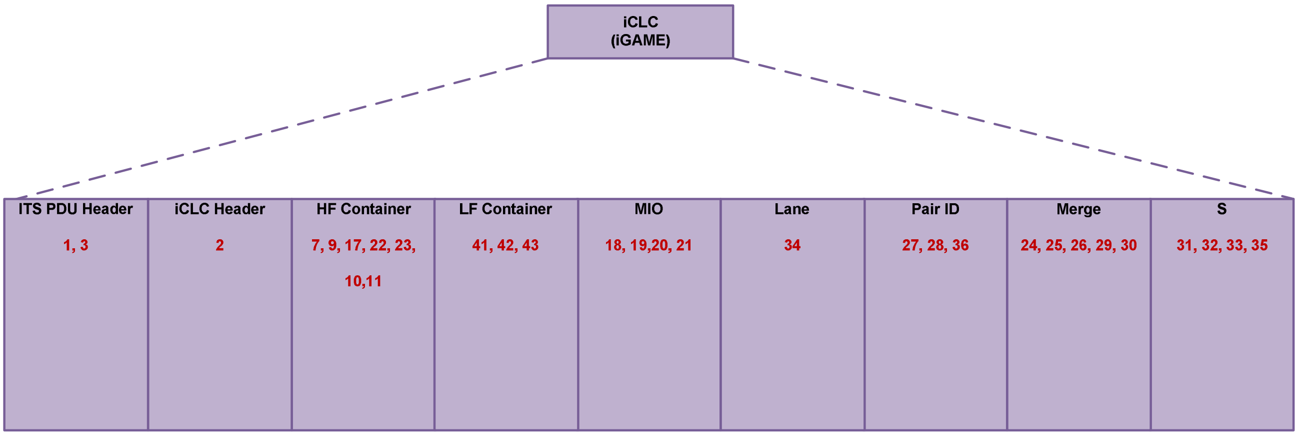

Figure 14 General structure of iCLCM format

So this structure is used for the iCLCM ASN.1 definition which is available in Appendix C.

In addition to the message set definition, there is also impact at the facility layer, see the information in the table below taken from i-GAME D3.1 [1]. This is related to the GeoNetworking (GN) forwarding mechanism and the BTP port number to use.

Table 3 BTP port numbers

Well-known BTP port	ITS facility entity	Comments
2001	CAM	ETSI EN 302 637-2, but used on non-standard frequencies 1-25 Hz, use GN single hop broadcast
2002	DENM	ETSI EN 302 637-3, use GN geobroadcast
2010	i-GAME Cooperative Lane Change Message (iCLCM)	CAM-like message with 1-25 Hz update frequency, use GN single hop broadcast

6 Definition of ASN.1 Libraries

6.1 Introduction

The messages within the C-ITS standards coming from facilities are defined in Abstract Syntax Notation 1 or ASN.1 format. This notation describes the rules for decoding, encoding, and the structures of the messages. It facilitates the exchange of structured data in a way to become independent of machine architecture and implementation language. In addition, ETSI TS 102 894-2 [8] describes the Application and Facilities layer common data dictionary. In this technical specification a set of data types is identified that is commonly used in multiple ITS applications and facilities layer messages. A common data dictionary is therefore defined. For each data type, this common dictionary includes a textual description of the semantic of the data types. It also includes the ASN.1 definition of the data type. Therefore, this common data dictionary can be imported by any message when necessary during the encoding and decoding procedure.

i-GAME defines the messages needed for the GCDC2016 scenarios in ASN.1 format. The teams can use ASN.1 compilers to generate the i-GAME message set libraries for the specific architecture and operating systems of the Communication Unit equipment they use in their vehicle. See Section 6.3 for more information on ASN.1 tools available for the teams. The iCLCM ASN.1 definition is available in Appendix C.

6.2 ASN.1 definitions

The ITS data dictionary is a repository that includes a list of data elements (DE) and data frames (DF) that represent data as well as information necessary for the realization of ITS applications and ITS facilities. The DE/DF can be used to construct CAM and DENM messages. According to the usage purpose, a DE or a DF can be classified into the categories Message Management or Application usage. Each DE and DF is defined by a set of attributes, enabling the identification of the data in question.

The CAM format as all the messages we use are presented in ASN.1. Unaligned Packed Encoding Rules (PER) as defined in Recommendation ITU-T X.691/ISO/IEC 8825-2 [9] is used for CAM encoding and decoding.

6.3 ASN.1 Tooling available for the teams

As mentioned in the previous section, with the i-GAME ASN.1 message definitions, teams are able to compile their own software libraries for their specific communication platform. In addition, organizers might distribute a binary library for C, C++ and/or Java to convert some (predefined) data structure into a UPER-encoded byte string and back, to remove the need to work directly with ASN.1.

Tools for ASN.1 are available on almost all operating systems. They generate code for popular programming languages such as Java, C and C++. There are tools that have been ported to over 250 different computing platforms. There are a lot of well-tested ASN.1 tools that have been used for a long time. See the ITU-T web page[1] for some ASN.1 compilers, several open source and commercial software is available, some examples below:

- ASN.1 Encoder and Decoder for Unaligned Packed Encoding Rules (UPER, ITU-T Recommendation X.691 | ISO/IEC 8825-2). Encoder encodes objects of classes annotated with asn1-datatypes[2].
- An open source ASN.1 compiler is asn1c[3]. This tool creates C source code and is tied to gcc-specific extensions in several key places.

[1] ITU-T ASN.1 projects: http://www.itu.int/en/ITU-T/asn1/Pages/Tools.aspx
[2] Available at: https://github.com/alexvoronov/geonetworking/tree/master/asn1-uper
[3] available at https://github.com/vlm/asn1c

- An online ASN.1 compiler at http://lionet.info/asn1c which generates platform-independent code.

OSS Nokalva[4] is sponsoring the GCDC2016 teams to use their commercial compiler and runtime to work with ASN.1 in Java. It is up to the teams to decide if they want to use these tools. More information on this software and how to use it will be available via the i-GAME website in the registered teams section.

[4] See website: http://www.oss.com/asn1/products/asn1-products.html

7 Verification of interaction message sets

Some tooling will be provided by the i-GAME organization, available for the teams to test their individual implementations. It is up to the teams to decide on usage of the test tools.

7.1 I-GAME Simulation Test Tool

The i-GAME Simulation Test Tool (iSTT) allows participants of the GCDC 2016 to test (many of) the communication and controller aspects of their equipment well in advance of the actual event.

The basic idea is to offer an online platform to connect through the use of tunnelling. The wireless data transmitted and received normally on a wireless interface is redirected over Internet to a virtual network. By this the communications and interactions needed in the scenarios can be tested. In addition, controller functionality can be tested by means of locally deployed software component. This tooling can help the teams to prepare for the challenge without requiring physical presence. The iSTT consists of two functional entities as depicted in Figure 15 below:

- Interaction Test Tool (ITT): A software component instantiated on a global IPv4 reachable through the global Internet.
- Vehicle Simulator (VeS): A software component instantiated locally by the Participant for testing his vehicle controllers.

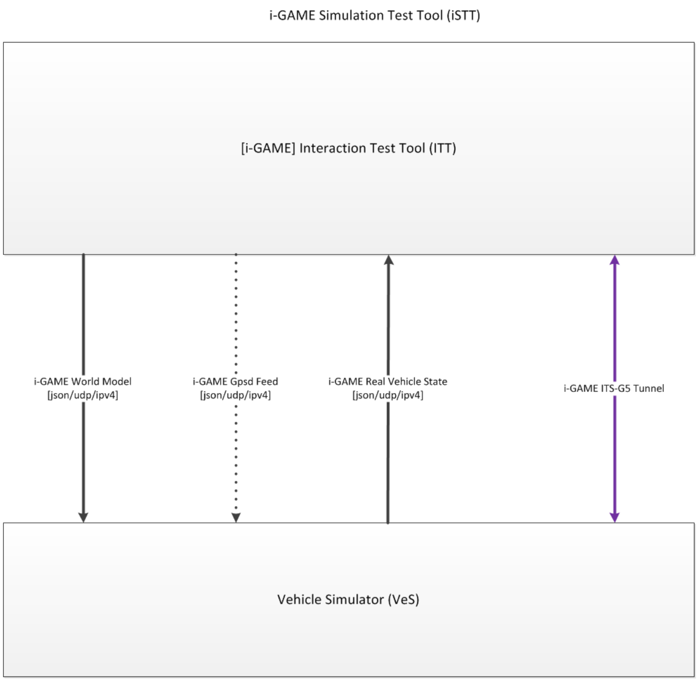

Figure 15: Overview of the i-GAME Simulation Test Tool.

The online ITT is the part on which the teams can connect to the Vehicle Simulator (VeS) via a VPN tunnel and test their own implementations and communication set-up. The use of this tool set is optionally for the teams. For more details see i-GAME D2.3 [10] and the information related to the i-GAME D3.3 Online software platform for interoperability testing of interaction [11].

List of references

[1] i-GAME D3.1 Wireless communication basic specification document

[2] i-GAME D2.1 Interaction protocol

[3] i-GAME D1.3 Functional Architecture document

[4] i-GAME D1.1 Specifications of scenarios document

[5] ETSI EN 302 637-2 ITS Vehicular Communications: Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service, v1.3.2 (2014-11)

[6] ETSI EN 302 637-3 ITS Vehicular Communications: Basic Set of Applications; Part 3: Specification of Decentralized Environmental Notification Basic Service, v1.2.2 (2014-11)

[7] Autonet2030 Specifications for the enhancement to existing LDM and cooperative communication protocol standards

[8] ETSI TS 102 894-2 Intelligent Transport Systems (ITS); Users and applications requirements; Part 2: Applications and facilities layer common data dictionary, v1.2.1, (2014-09)

[9] Recommendation ITU-T X.691/ISO/IEC 8825-2 (1997-12): "Information technology - ASN.1 encoding rules: Specification of Packed Encoding Rules (PER)"

[10] i-GAME D2.3 Simulation toolset for evaluation of supervisory control systems for cooperative driving

[11] i-GAME D3.3 Online software platform for interoperability testing of interaction

[12] TISA specification TAWG11071 (2011-11-07, drafted to potentially become ISO/TS 21219 Part 15): "Intelligent Transport Systems (ITS) - Traffic and Travel Information (TTI) via Transport Protocol Experts Group, Generation 2 (TPEG2) - Part 15: Traffic Event Compact (TPEG2-TEC-3.1/001)"

[13] ISO 1176:1990 “Road vehicles – Masses – Vocabulary and codes”.

[14] ETSI TS 101 539-1: “Intelligent Transport Systems (ITS); V2X Applications; Part 1: Road Hazard Signalling (RHS) application requirements specification”.

[15] ISO 3779 (2011-07): “Road vehicles – Vehicle identification number (VIN) Content and structure”.

[16] SAE J2735 (2009-11-19): “Dedicated Short Range Communications (DSRC) Message Set Dictionary”.

[17] I-GAME D2.5 Algorithm-protocol for the transition of control

[18] ETSI TS 101 539-3 Longitudinal Collision Risk Warning (LCRW)

[19] ETSI TS 101 539-2 Intersection Collision Risk Warning (ICRW)

[20] ISO 8601 “Data elements and interchange formats — Information interchange — Representation of dates and times”

[21] ETSI TS 101 539-1 Road Hazard Signalling (RHS)

[22] I-GAME D4.5 Safety analysis of scenarios and requirements.

[23]I-GAME D2.2 Generic real-time control system

Appendix A Description of Data Elements and Data Frames for i-GAME scenarios

A.1.1 header

Description	Message identifier, the common message header for application and facilities layer messages. It is included at the beginning of an ITS message as the message header
Data Format	ItsPduHeader
Range	{0,…,255}
Resolution	-
Remarks	Message type identifier, comparable with CAM header message ID: 1: DENM 2: CAM 10: iCLCM messageID INTEGER{denm(1),cam(2), poi(3), spat(4), map(5), ivi(6), ev-rsr(7)} (0..255), iCLCM(10)

A.1.2 GenerationDeltaTime

Description	Generation time of the (CAM) message in milliseconds
Range	{0,…,65535}
Resolution	1 milliseconds
Remarks	Time corresponding to the time of the reference position in the CAM, considered as time of the CAM generation. The value of the DE shall be wrapped to 65 536. This value shall be set as the remainder of the corresponding value of TimestampIts divided by 65 536 as below: generationDeltaTime = TimestampIts mod 65 536 TimestampIts represents an integer value in milliseconds since 2004-01-01T00:00:00:000Z as defined in ETSI TS 102 894-2 [8].

A.1.3 Station ID

Description	Station unique identifier, identifier for an ITS-S
Range	{0,…,4294967295}
Resolution	-
Remarks	Station identifier, comparable with CAM header Station ID of the originating ITS-S. Station = object vehicle

A.1.4 Station Type

Description	Station Type definition field
Range	{0,…,255} 0: unknown (invisible) 1-255: visible
Resolution	-
Remarks	unknown(0), pedestrian(1), cyclist(2), moped(3), motorcycle(4), passengerCar(5), bus(6), lightTruck(7), heavyTruck(8), trailer(9), specialVehicles(10), tram(11), roadSideUnit(15)} (0..255)

	As The type of an ITS-S. The station type depends on the integration environment of ITS-S into vehicle, mobile devices or at infrastructure. specialVehicles(10) can be used for Emergency Vehicles to indicate an Emergency Vehicle in operation. roadSideUnit(15) available for RSUs Use eg. EmulatedRSU(16) for an OPC emulating a RSU

A.1.5 Vehicle Role

Description	Vehicle Role of the station, played by a vehicle at a certain point in time
Range	{0,...,255} 0: default 1 - 255: visible
Resolution	-
Remarks	The role of the vehicle ITS-S that originates the message at a point in time. VehicleRole {default(0), publicTransport(1), specialTransport(2), dangerousGoods(3), roadWork(4), rescue(5), emergency(6), safetyCar(7), agriculture(8),commercial(9),military(10),roadOperator(11),taxi(12), reserved1(13), reserved2(14), reserved3(15)} For EV in CAM an emergency(6) Vehicle Role can be used and a emergencyContainer must be included in the special vehicle container.

A.1.6 Vehicle length

Description	This DF includes: • vehicleLengthValue: Vehicle length of the vehicle ITS-S that originates the CAM. If there are vehicle attachments like a trailer, or overhanging attachments like a crane, that extend the vehicle length to the front and/or rear; then the vehicleLengthValue shall provide the length for the vehicle including the attachments. • vehicleLengthConfidenceIndication: indication of whether trailer is detected to be present and whether the length of the trailer is known.
Range	{0,...,1022}
Resolution	0.1 m
Remarks	Length of a vehicle. The value shall be set to 1 022 if the vehicle length is equal to or greater than 102,2 metres

A.1.7 Vehicle rear axle location

Description	Vehicle rear axle location with respect to the geometrical centre of the vehicle
Range	{0,...,4095}
Resolution	0.01 m
Remarks	Location of the rear axle with respect to the vehicle reference position as the perpendicular distance between the vehicle front line of the bounding box (front bumper) and the front wheel axle. This is used for coordinate transformation, because this is a reference point for multiple sensors locations and sensor fusion. non-CAM data

A.1.8 Vehicle width

Description	Vehicle width, measured of the vehicle ITS-S that originates the CAM, including side mirrors
Range	{0,...,61}
Resolution	0.01 m
Remarks	Width of a vehicle, including side mirrors. For a vehicle width equal to or greater than 6.1 metres, the value shall be set to 61. The value shall be set to 62 if the information is unavailable.

A.1.9 Controller type

Description	Controller type
Range	{0,…,3}
Resolution	-
Remarks	0: Manual 1: CC 2: ACC 3: CACC Used in both scenarios. Also a virtual CACC is mentioned in scenario 2 comparable with 3: CACC only a virtual vehicle is used for the CACC.

A.1.10 Vehicle response time constant

Description	Vehicle response time constant
Range	{0..1001}
Resolution	0.01 s
Remarks	Vehicle characteristics, value determined by the drive-line specifications of the vehicle. Use 1001 when information is unavailable. Non-CAM

A.1.11 Vehicle response time delay

Description	Vehicle response time delay
Range	{0,…,1001]
Resolution	0.01 s
Remarks	Vehicle characteristics, value determined by the drive-line specifications of the vehicle. Use 1001 when this information is unavailable. Non-CAM

A.1.12 referencePosition

Description	The geographical position of a position or of an ITS-S. It represents a geographical point position.
Data Format	DF_referencePosition
Remarks	The DF shall include the following information: • latitude: latitude of the geographical point; it shall be presented as specified in clause A.41 Latitude [8]. • longitude: longitude of the geographical point; it shall be presented as specified in clause A.44 Longitude [8].

	• positionConfidenceEllipse: accuracy of the geographical position; it shall be presented as specified in clause A.119 PosConfidenceEllipse [8]. • altitude: altitude and altitude accuracy of the geographical point; it shall be presented as specified in clause A.103 Altitude [8].

A.1.12.1 Latitude reference position

Description	Vehicle Latitude Reference position
Data Element	DE_Latitude
Range	{-900000000,...,900000001}
Resolution	0.1 microdegrees
Remarks	Absolute geographical latitude in a WGS84 coordinate system, providing a range of 90 degrees in north or in south hemisphere. Positive values are used for latitude in north of the Equator, negative values are used for latitude in south of the Equator. When the information is unavailable, the value shall be set to 900 000 001. • latitude: latitude of the geographical point; Absolute geographical latitude in a WGS84 coordinate system, Latitude ::= INTEGER {oneMicrodegreeNorth (10), oneMicrodegreeSouth (-10), unavailable(900000001) } (-900000000..900000001) Reference position used is middle front bumper of vehicle at GenerationDeltaTime.

A.1.12.2 Longitude reference position

Description	Vehicle Longitude Reference position
Range	{-1800000000,...,1800000001}
Data Element	DE_Longitude
Resolution	0.1 microdegree
Remarks	Absolute geographical longitude in a WGS84 co-ordinate system, providing a range of 180 degrees to the east or to the west of the prime meridian. Negative values are used for longitudes to the west, positive values are used for longitudes to the east. When the information is unavailable, the value shall be set to 1 800 000 001. • longitude: longitude of the geographical point; Absolute geographical longitude in a WGS84 coordinate system, Longitude ::= INTEGER {oneMicrodegreeEast (10), oneMicrodegreeWest (-10), unavailable(1800000001) } (-1800000000..1800000001) Reference position used is middle front bumper of vehicle at GenerationDeltaTime.

A.1.12.3 Position confidence ellipse interval 95%

Description	Vehicle Position confidence interval 95% DF that provides the horizontal position accuracy in a shape of ellipse with a predefined confidence level (e.g. 95 %). The centre of the ellipse shape corresponds to the reference position point for which the position accuracy is evaluated
Data Format	DF_PosConfidenceEllipse
Range	{0,...,4095}
Resolution	-1: accuracy ≤ 1cm. n (n > 1 and n < 4 093): accuracy ≤ n cm.

	4 093: accuracy ≤ 4 093 cm. 4 094: accuracy > 4 093 cm. 4 095: accuracy information unavailable.
Remarks	Accuracy of the geographical position; it provides the horizontal position accuracy in a shape of ellipse with a predefined confidence level (e.g. 95 %). The centre of the ellipse shape corresponds to the reference position point for which the position accuracy is evaluated. It include the following information: • semiMajorConfidence: half of length of the major axis, i.e. distance between the centre point and major axis point of the position accuracy ellipse. • semiMinorConfidence: half of length of the minor axis, i.e. distance between the centre point and minor axis point of the position accuracy ellipse. • semiMajorOrientation: orientation direction of the ellipse major axis of the position accuracy ellipse with regards to the WGS84 north.

A.1.13 Vehicle heading

Description	Vehicle heading
Data Format	DF_Heading
Remarks	Heading in a WGS84 co-ordinates system. The DF shall include the following information: • headingValue: a heading value. It shall be presented as defined in clause A.35 HeadingValue [8]. • headingConfidence: the accuracy of the reported heading value with a predefined confidence level. It shall be presented as defined in clause A.34 HeadingConfidence [8].

A.1.13.1 HeadingValue

Description	Vehicle heading
Data Element	DE_HeadingValue
Range	WGS84 co-ordinates system {0,...,3061}
Resolution	0.1 degree
Remarks	Orientation of a heading with regards to the WGS84 north. 0: wgs84North 900: wgs84East 1800: wgs84South 2700: wgs84West 3061: Unavailable When the information is not available, the DE shall be set to 3 601.

A.1.13.2 Heading confidence

Description	Heading confidence interval 95%
Data Element	DE_HeadingConfidence
Range	{1,...,127}
Resolution	0.1 degree
Remarks	The heading accuracy shall provide the accuracy of the measured vehicle heading with a confidence level of 95 %. Otherwise, the value of the heading Confidence shall be set to unavailable.

	The value shall be set to: • 1 if the heading accuracy is equal to or less than 0,1 degree, • n (n > 1 and n < 125) if the heading accuracy is equal to or less than n × 0,1 degree, • 125 if the heading accuracy is equal to or less than 12,5 degrees, • 126 if the heading accuracy is out of range, i.e. greater than 12,5 degrees, • 127 if the heading accuracy information is not available.

A.1.14 Speed

Description	Longitudinal vehicle velocity
Data Format	DF_Speed
Range	{0,...,16383}
Resolution	0.01 m/s
Remarks	It describes the speed and corresponding accuracy of the speed information for a moving object (e.g. vehicle). The DF shall include the following information: • speedValue: speed value. It shall be presented as defined in clause A.74 SpeedValue [8]. • speedConfidence: accuracy of the reported speed value. It shall be presented as defined in clause A.72 SpeedConfidence [8]. Reference position used is middle front bumper of vehicle at GenerationDeltaTime.

A.1.14.1 SpeedValue

Description	Longitudinal vehicle velocity
Data Element	DE_SpeedValue
Range	{0,...,16383}
Resolution	0.01 m/s
Remarks	A speed value. For values equal to or greater than 163,82 m/s, the value shall be set to 16 382. When the information is not available, the value shall be set to 16 383. Reference position used is middle front bumper of vehicle at GenerationDeltaTime.

A.1.14.2 Speed confidence

Description	Longitudinal vehicle velocity confidence interval 95%
Data Element	DE_SpeedConfidence
Range	{1,...,127}
Resolution	0.01 m/s
Remarks	The absolute accuracy of a speed value information for a predefined confidence level. The required confidence level is defined by the station applying this. The value shall be set to: • 1 if the speed accuracy is equal to or less than 1 cm/s. • n (n > 1 and n < 125) if the speed accuracy is equal to or less than n cm/s. • 125 if the speed accuracy is equal to or less than 125 cm/s. • 126 if the speed accuracy is out of range, i.e. greater than 125 cm/s. • 127 if the speed accuracy information is not available.

A.1.15 YawRate

Description	Vehicle yaw rate
Data Format	DF_YawRate
Remarks	Yaw rate of vehicle at a point in time. The DF shall include the following information: • yawRateValue: yaw rate value at a point in time. It shall be presented as defined in clause A.101 YawRateValue of Common Data Dictionary [8]. • vehicleLengthConfidenceIndication: accuracy of reported yaw rate value. It shall be presented as defined in clause A.100 YawRateConfidence [8].

A.1.15.1 YawRateValue

Description	Vehicle yaw rate
Data Element	DE_YawRateValue
Range	{-32767,...,32767}
Resolution	0.01 degree per second
Remarks	YawRate_Value: Yaw rate of vehicle at a point in time. The leading sign denotes the direction of rotation. Positive values indicate that the rotation is anti-clockwise (i.e. to the left). The value shall be set to 32 766 if the yaw rate is equal to or greater than 327,66 degrees/second to the left. Negative values indicate that the rotation is clockwise (i.e. to the right). The value shall be set to -32 766 if the yaw rate is equal to or greater than 327,66 degrees/second to the right. The value shall be set to 32 767 if the information is not available VehicleYawRate ::= INTEGER {straight(0), degSec-000-01ToRight(-1), degSec-000-01ToLeft(1), unavailable(32767)} (-32766..32767)

A.1.15.2 Yaw rate confidence interval 95%

Description	Vehicle yaw rate confidence interval 95%
Data Element	DE_YawRateConfidence
Range	{0,...,8}
Resolution	-
Remarks	This denotes the absolute accuracy range for reported yaw rate value for a predefined confidence level (e.g. 95 %). The required confidence level is defined by the originating station. The value shall be set to: • 0 if the accuracy is equal to or less than 0,01 degree/second • 1 if the accuracy is equal to or less than 0,05 degrees/second • 2 if the accuracy is equal to or less than 0,1 degree/second • 3 if the accuracy is equal to or less than 1 degree/second • 4 if the accuracy is equal to or less than 5 degrees/second • 5 if the accuracy is equal to or less than 10 degrees/second • 6 if the accuracy is equal to or less than 100 degrees/second • 7 if the accuracy is out of range, i.e. greater than 100 degrees/second • 8 if the accuracy information is unavailable YawRateConfidence ::= ENUMERATED { degSec-000-01 (0), degSec-000-05 (1), degSec-000-10 (2), degSec-001-00 (3), degSec-005-00 (4), degSec-010-00 (5), degSec-100-00 (6), outOfRange (7), unavailable (8)

A.1.16 Longitudinalacceleration

Description	Longitudinal vehicle acceleration
Data Format	DF_LongitudinalAcceleration
Remarks	It indicates the vehicle acceleration at longitudinal direction and the accuracy of the longitudinal acceleration. The DF shall include: • longitudinalAccelerationValue: longitudinal acceleration value at a point in time. It shall be presented as specified in clause A.45 LongitudinalAccelerationValue [8]. • longitudinalAccelerationConfidence: accuracy of the reported longitudinal acceleration value with a predefined confidence level. It shall be presented as defined in clause A.1 AccelerationConfidence [8].

A.1.16.1 LongitudinalAccelerationValue

Description	Longitudinal vehicle acceleration value
Data Element	DE_LongitudinalAccelerationValue
Range	{-160,...,161}
Resolution	0.1 m/ s^2
Remarks	Vehicle acceleration at longitudinal direction in the centre of the mass of the empty vehicle. It corresponds to the vehicle coordinate system as specified in ISO 8855 [2]. Negative values indicate that the vehicle is braking. For values equal to or greater than 16 m/ s^2, the value shall be set to -160. Positive values indicate that the vehicle is accelerating. For acceleration equal to or greater than 16 m/ s^2 the value shall be set to 160. When the data is unavailable, the value shall be set to 161. Reference position used is middle front bumper of vehicle at GenerationDeltaTime.

A.1.16.2 longitudinalAccelerationConfidence

Description	Longitudinal vehicle acceleration confidence interval 95%
Data Element	DE_AccelerationConfidence
Range	{1,...,102}
Resolution	0.1 m/ s^2
Remarks	The absolute accuracy of a reported vehicle acceleration value with a predefined confidence level (e.g. 95 %). The required confidence level is defined by the corresponding standards applying the DE. The value shall be set to: • 1 if the acceleration accuracy is equal to or less than 0,1 m/s2. • n (n > 1 and n < 100) if the acceleration accuracy is equal to or less than n × 0,1 m/s2. • 100 if the acceleration accuracy is equal to or less than 10 m/s2. • 101 if the acceleration accuracy is out of range i.e. greater than 10 m/s2. • 102 if the data is unavailable.

A.1.17 Desired longitudinal vehicle acceleration

Description	Desired longitudinal vehicle acceleration
Range	{-1000,…,1001}
Resolution	0.01 m/ s^2
Remarks	Set-point of control system, 1001 when unavailable. Non-CAM

A.1.18 MIO ID (measured by object vehicle)

Description	Most import object identifier (measured by object vehicle)
Range	{0,...,4294967295}
Resolution	-
Remarks	StationID of most important object in front of the vehicle Non-CAM. 0: no ID

A.1.19 MIO range (measured by object vehicle)

Description	MIO range (as measured by object vehicle)
Range	{0,...,65535}
Resolution	0.01 m
Remarks	Distance to MIO in front of the vehicle, 65535 when unavailable. Non-CAM

A.1.20 MIO bearing (measured by object vehicle)

Description	MIO bearing (as measured by object vehicle)
Range	(-1571,…,1572)
Resolution	0.002 rad
Remarks	1572 when unavailable. Non-CAM

A.1.21 MIO range rate (measured by object vehicle)

Description	MIO range rate (as measured by object vehicle)
Range	{-32767,...,32767}
Resolution	0.01 m/s
Remarks	Velocity of the MIO, 32767 when unavailable. Non-CAM

A.1.22 Time headway

Description	Time headway
Range	{0..361} s
Resolution	0.1 s
Remarks	Own vehicle time gap to the vehicle in front, 361 when unavailable. Non-CAM

A.1.23 Cruise speed

Description	Cruise speed
Range	{0,...,5001}
Resolution	0.01 m/s
Remarks	Nominal speed of platoon, coming from the first vehicle (FV)/platoon leader, 5001

	when unavailable. Non-CAM

A.1.24 Merge request flag

Description	Merge request flag
Range	{0,1}
Accuracy	-
Remarks	0: default, no merge requested 1: merge requested Used by vehicle to request a merging manoeuvre.

A.1.25 Safe-to-merge (STOM) flag

Description	Safe-to-merge (STOM) flag
Range	{0,1}
Resolution	-
Remarks	0: not safe 1: safe Use to indicate that it is safe to start a merging manoeuvre. Non-CAM

A.1.26 Merging flag

Description	Merging flag
Range	{0,1}
Resolution	-
Remarks	Merging status of vehicle: gaps are made and indicate the vehicle is ready for merging. 0: not Merge ready 1: Merge ready Non-CAM

A.1.27 ID of fwd pair partner

Description	Identifier of the forward pair partner. The station ID of the vehicle is used as the unique identifier.
Range	{0,...,4294967295}
Resolution	-
Remarks	StationID of the forward pairing partner based on its station ID. Non-CAM

A.1.28 ID of bwd pair partner

Description	Identifier of the backward pair partner. The station ID of the vehicle is used as the unique identifier.
Range	{0..4294967295}
Resolution	-
Remarks	Station ID of the backward pairing partner based on its station ID. Non-CAM

A.1.29 Tail vehicle flag

Description	Tail vehicle flag
Range	{0,1}
Resolution	-
Remarks	Last vehicle in the platoon. 0 = default 1 = Last Vehicle Non-CAM

A.1.30 Head vehicle flag

Description	Head vehicle flag
Range	{0,1}
Resolution	-
Remarks	Identifier of the first vehicle (FV) in the platoon. 0 = default, not first vehicle 1 = First Vehicle Non-CAM

A.1.31 Platoon ID

Description	Platoon (or lane) identifier, used to assign a specific road section (left, right) to the platoon
Range	{0,...,255}
Resolution	-
Remarks	1=platoon A 2=platoon B 3=not used

A.1.32 Travelled distance inside the CZ

Description	Travelled distance of the vehicle inside the CZ
Range	{0,..,10000}
Resolution	0.1 m
Remarks	The range for this also depends on the intersection geometry. The value depends on the reference point and the coordinate transformation used. The details how to calculate this distance are available in D2.2 document, [23] Non-CAM

A.1.33 Intention

Description	Intention of the vehicle travelling in the CZ
Range	{1,2,3}
Resolution	-
Remarks	1 Straight, no turning 2 Turn left 3 Turn right

	Non-CAM

A.1.34 Lane on which the vehicle enters the CZ

Description	Lane on which the vehicle enters the CZ
Range	{1,2,3,4}
Resolution	-
Remarks	The lanes corresponds to the vehicle number in Figure 3 of Section 2.3. The predefined lane numbers are available in D2.1 [2] and also related information is available in D2.2 [23]; Use 4 when this information is not available. Non-CAM

A.1.35 Intersection vehicle counter

Description	Intersection vehicle counter. Number (m in D2.1, [2]) assigned to vehicle when entering the CZ to perform Target Vehicle Assignment (TVA). This is a counter of the vehicles inside the CZ
Range	{0,1,2,3}
Resolution	-
Remarks	Number m=0 when no vehicles inside the CZ. Every vehicles that enters the CZ is assigned a distinct value of m. Used in intersection scenario with a maximum of 3 vehicles. This corresponds to the vehicle number in Figure 3 of Section 2.3. Non-CAM

A.1.36 Pair acknowledge Flag

Description	Pair acknowledge flag
Range	{0,1} 0 = not acknowledged 1 = acknowledged
Resolution	-
Remarks	Currently not used. Can be used to enhance the reliability or fail-safety of the interaction protocol in the merging scenario.

A.1.37 Reference Time

Description	This data element refers to the time at which a new DENM, an update DENM or a cancellation DENM is generated.
Data Element	DE_TimestampIts
Range	{0,...,4398046511103}
Resolution	milliseconds
Remarks	Number of milliseconds since 2004-01-01T00:00:00.000Z

A.1.38 EventType

Description	Description for the event type, including direct cause and sub cause This DE shall be presented as specified in ETSI TS 102 894-2 DF_CauseCode [8]
Data format	DF_CauseCode
Remarks	The DF shall include the following information: • causeCode: the type of a direct cause of a detected event. It shall be

	presented as defined in clause A.10 CauseCodeType, • subCauseCode: sub type of the direct cause. It shall be presented as defined in clause A.81 SubCauseCodeType

A.1.38.1 causeCode

Description	Description for the event type, the direct cause
Data Element	DE_CauseCodeType
Range	{0,...,255}
Resolution	-
Remarks	Of importance for executing the scenarios are: roadworks (3), stationaryVehicle (94), emergencyVehicleApproaching (95) dangerousSituation (99)

A.1.38.2 SubCauseCodeType

Description	Description for the event type, the sub cause, related to the direct cause
Data Element	DE_SubCauseCodeType
Range	{0,...,255}
Resolution	-
Remarks	Of importance for executing the scenarios are: roadworks (3): subcause (0), unknown) stationaryVehicle (94): subCause (2) vehicle Breakdown emergencyVehicleApproaching (95): subcause (1) EV approaching dangerousSituation (99): subcause (1) Emergency electronic brake light

A.1.39 closedLanes

Description	Indicates whether the roadworks has casued the closure of one or several driving lanes
Data Element	DF_ClosedLanes
Remarks	The DF shall include the following information: • hardShoulderStatus: this information is optional and shall be included if the information is known. It indicates the open/closing status of hard shoulder lanes. It shall be presented as defined in clause A.29 HardShoulderStatus [8], (not used in the scenario) • drivingLaneStatus: it indicates the open/closing status of driving lanes. It shall be presented as defined in clause A.23 DrivingLaneStatus [8].

A.1.39.1 drivingLaneStatus

Description	DE that indicates whether a driving lane is open to traffic
Data Element	DE_DrivingLaneStatus
Range	{1,...,14}
Resolution	-
Remarks	A lane is counted from outside boarder of the road. The numbering is matched to LanePosition DE as defined in A.1.40. If a lane is closed to traffic, the

	corresponding bit shall be set to 1. Otherwise, it shall be set to 0

A.1.40 LanePosition

Description	The lane position of the event position in the road counted from the outside boarder of the road. This DE is included in the alacarte container to be used in combination with Emergency Vehicle Aproaching
Data Element	DE_LanePosition
Range	{-1,...,14}
Resolution	-
Remarks	LanePosition ::= INTEGER {offTheRoad(-1), hardShoulder(0), outermostDrivingLane(1), secondLaneFromOutside(2)} (-1..14)

A.1.41 Participants ready

Description	Participants ready message
Range	{0,1}
Resolution	-
Remarks	Test message used in the preparation phase, before the competition starts. With the participating vehicles start in their standby / wait state upon receiving this message from the RSU. First to check that communications are online. And also meaning: switch on the controller, choose the lane, choose the scenario (merging/intersection/EV). 0 = not ready 1 = ready

A.1.42 Start scenario

Description	Start Platoon X
Range	{0, 1}
Resolution	-
Remarks	This message is used in the preparation phase, before the competition starts. A RSU send start message for platoon X / OPC X to start driving and accelerating towards a certain speed. Two static options are available: 0 = start platoon A at speed 80km/h 1 = start platoon B at speed 60 km/h RSU send start message for platoon B / OPC B to start driving and accelerating towards a speed of 60km/h From RSU/OPC A Platoon A receives their start message and starts driving and accelerating towards a speed of 80km/h

A.1.43 EoS

Description	End of Scenario / end of judging criteria
Range	{1}
Resolution	-
Remarks	This message when present, is used in notify the contestants that the judging part of the scenario is ended

A.1.44 Reserve/spare/future use

Description	Reserve/spare/future use
Range	-
Resolution	-
Remarks	tbd

A.1.45 Reserve/spare/future use

Description	Reserve/spare/future use
Range	-
Resolution	-
Remarks	tbd

A.1.46 Reserve/spare/future use

Description	Reserve/spare/future use
Range	-
Resolution	-
Remarks	tbd

Appendix B Mapping of messages onto the scenarios

Message ID	Message description	Scenario
1	Header	1, 2, 3
2	GenerationDeltaTime	1, 2, 3
3	Station ID	1, 2, 3
4	Station Type	1, 2, 3
5	Vehicle Role	1, 2, 3
6	Vehicle length	1,2
7	Vehicle rear axle location	1,2
8	Vehicle width	1, 2, 3
9	Controller type	1, 2, 3
10	Vehicle response time constant	1,2
11	Vehicle response time delay	1,2
12	ReferencePosition (latitude, longitude, confidence)	1, 2
13	Heading (Heading, confidence)	1, 2
14	Speed	1, 2
15	YawRate	1, 2
16	Longitudinal vehicle acceleration	1, 2
17	Desired longitudinal vehicle acceleration	1, 2
18	MIO ID (measured by object vehicle)	1
19	MIO range (measured by object vehicle)	1
20	MIO bearing (measured by object vehicle)	1
21	MIO range rate (measured by object vehicle)	1
22	Time headway	1
23	Cruise speed	1
24	Merge request flag	1
25	Safe-to-merge (STOM) flag	1
26	Merging flag	1
27	ID of fwd pair partner	1
28	ID of bwd pair partner	1
29	Tail vehicle flag	1
30	Head vehicle flag	1
31	Platoon ID	1
32	Travelled distance inside the CZ	2
33	Intention (left, right, or straight)	2
34	Lane on which the vehicle enters the CZ	1,2
35	Intersection vehicle counter	2
36	Pair acknowledge flag	1
37	Reference Time	3
38	EventType (Roadworks, Stationary vehicle, Emergency vehicle approaching, Dangerous Situation (Emergency electronic brake light)	1,3
39	ClosedLanes	1
40	LanePosition	3

Message ID	Message description	Scenario
41	Participants ready	1,2,3
42	Start scenario	1,2,3
43	EoS (End of Scenario)	1,2,3
44	Reserve/spare/future use	Tbd
45	Reserve/spare/future use	Tbd
46	Reserve/spare/future use	tbd

Appendix C ASN.1 specification of i-GAME messages

```
ICLCM DEFINITIONS AUTOMATIC TAGS ::=

BEGIN
-- The root data frame for iGAME Cooperative Lane Change Message
IGAMECooperativeLaneChangeMessage ::= SEQUENCE {
      itsHeader ItsPduHeader,
      iclcm IGAMECooperativeLaneChange-MessageBody
}

ItsPduHeader ::= SEQUENCE {
      protocolVersion ProtocolVersion,
      messageID MessageID,
      stationID StationID
}

IGAMECooperativeLaneChange-MessageBody ::= SEQUENCE {
      generationDeltaTime GenerationDeltaTime,
      iclcmParameters IclcmParameters
}

IclcmParameters ::= SEQUENCE {
      vehicleContainerHighFrequency VehicleContainerHighFrequency,
      vehicleContainerLowFrequency VehicleContainerLowFrequency OPTIONAL,
      mostImportantObjectContainer MostImportantObjectContainer,
      laneObject LaneObject,
      pairIdObject PairIdObject,
      mergeObject MergeObject,
      scenarioObject ScenarioObject
}

VehicleContainerHighFrequency ::= SEQUENCE {
      vehicleRearAxleLocation VehicleRearAxleLocation,
      controllerType ControllerType,
      vehicleResponseTime VehicleResponseTime,
      targetLongitudinalAcceleration TargetLongitudinalAcceleration,
      timeHeadway TimeHeadway,
      cruiseSpeed CruiseSpeed
}

VehicleContainerLowFrequency ::= SEQUENCE {
      participantsReady ParticipantsReady OPTIONAL,
      startPlatoon StartPlatoon OPTIONAL,
      endOfScenario EndOfScenario OPTIONAL
}

MostImportantObjectContainer ::= SEQUENCE {
      mioID StationID,
      mioRange MioRange,
      mioBearing MioBearing,
      mioRangeRate MioRangeRate
}

LaneObject ::= SEQUENCE {
```

```
      lane Lane
}

PairIdObject ::= SEQUENCE {
      forwardID StationID,
      backwardID StationID,
      acknowledgeFlag AcknowledgeFlag
}

MergeObject ::= SEQUENCE {
      mergeRequest MergeRequest,
      mergeSafeToMerge MergeSafeToMerge,
      mergeFlag MergeFlag,
      mergeFlagTail MergeFlagTail,
      mergeFlagHead MergeFlagHead
}

ScenarioObject ::= SEQUENCE {
      platoonID PlatoonID,
      distanceTravelledCZ DistanceTravelledCZ,
      intention Intention,
      counterIntersection CounterIntersection
}

VehicleResponseTime ::= SEQUENCE {
      vehicleResponseTimeConstant VehicleResponseTimeConstant,
      vehicleResponseTimeDelay VehicleResponseTimeDelay
}

--
-- ASN1-CAM,DENM shared header declaration
--
ProtocolVersion ::= INTEGER { currentVersion(1) } (0..255)
MessageID ::= INTEGER { denm(1), cam(2), iclcm(10) } (0..255)
StationID ::= INTEGER(0..4294967295)

GenerationDeltaTime ::= INTEGER { oneSecond(1000) } (0..65535)
VehicleRearAxleLocation ::= INTEGER { oneMeter(100) } (0..4095)
ControllerType ::= INTEGER { manual(0), cc(1), acc(2), cacc(3) } (0..3)
TargetLongitudinalAcceleration ::= INTEGER {
oneMeterPerSecondSquared(100), unavailable(1001) } (-1000..1001)
TimeHeadway ::= INTEGER { oneSecond(10), unavailable(361) } (0..361)
CruiseSpeed ::= INTEGER { oneMeterPerSecond(100), unavailable(5001) }
(0..5001)
ParticipantsReady ::= INTEGER { notReady(0), ready(1) } (0..1)
StartPlatoon ::= INTEGER { startPlatoonAAtSpeed80kph(0),
startPlatoonBAt60kph(1) } (0..1)
EndOfScenario ::= INTEGER { endOfScenario(1) } (1..1)
MioRange ::= INTEGER { oneMeter(100), unavailable(65535) } (0..65535)
MioBearing ::= INTEGER { zeroRadians(0), oneRadianRight(500),
unavailable(1572) } (-1571..1572)
MioRangeRate ::= INTEGER { zeroMeterPerSecond(0), oneMeterPerSecond(100),
unavailable(32767) } (-32767..32767)
Lane ::= INTEGER { laneOne(1), laneTwo(2), laneThree(3), unavailable(4) }
(1..4)
```

```
AcknowledgeFlag ::= INTEGER { notAcknowledged(0), acknowledged(1) }
(0..1)
MergeRequest ::= INTEGER { noMergeRequest(0), mergeRequest(1) } (0..1)
MergeSafeToMerge ::= INTEGER { notSafe(0), safe(1) } (0..1)
MergeFlag ::= INTEGER { notMergeReady(0), mergeReady(1) } (0..1)
MergeFlagTail ::= INTEGER { notLastVehicle(0), lastVehicle(1) } (0..1)
MergeFlagHead ::= INTEGER { notFirstVehicle(0), firstVehicle(1) } (0..1)
PlatoonID ::= INTEGER { platoonA(1), platoonB(2), notUsed(3) } (0..255)
DistanceTravelledCZ ::= INTEGER { oneMeter(10) } (0..10000)
Intention ::= INTEGER { straightNoTurning(1), turnLeft(2), turnRight(3) }
(1..3)
CounterIntersection ::= INTEGER { noVehicles(0), oneVehicle(1) } (0..3)
VehicleResponseTimeConstant ::= INTEGER { oneSecond(100),
unavailable(1001) } (0..1001)
VehicleResponseTimeDelay ::= INTEGER { oneSecond(100), unavailable(1001)
} (0..1001)

END
```