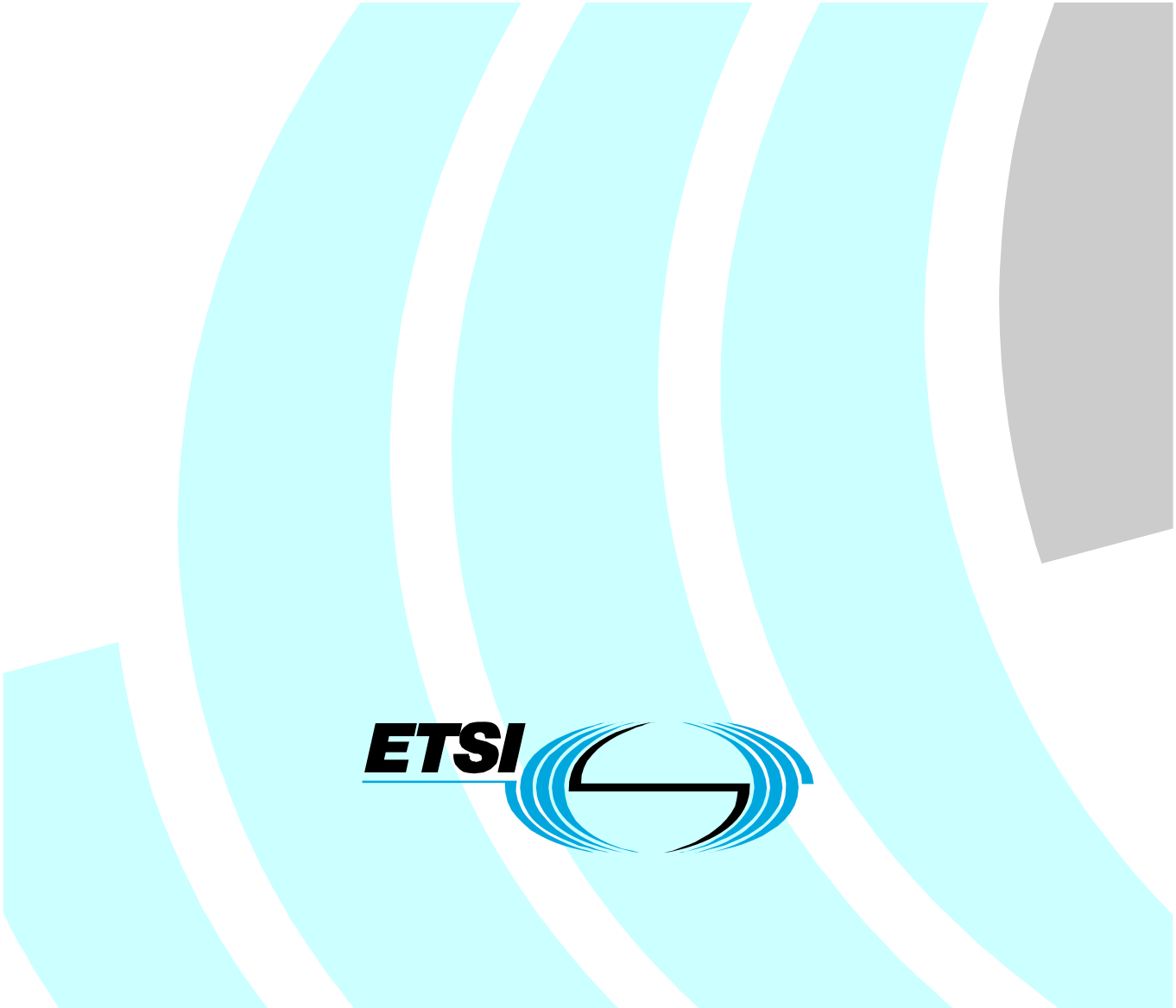
ETSI TR 102 698 V1.1.1 (2009-06)

Technical Report

Intelligent Transport Systems (ITS);
Vehicular Communications;
C2C-CC Demonstrator 2008;
Use Cases and Technical Specifications



Reference
DTR/ITS-0010003

Keywords
interoperability, ITS, testing, validation

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Intelligent Transport System (ITS).

In the present competitive environment, however, the risk of non-interoperability is increasing because of small windows of opportunity due to fast evolution of technology, or the use of non-open standards. But in large, complex systems such as those envisioned by ITS this is not workable in the long run.

It can be argued that interoperability is one of the main aims of ETSI standardization. Key components in the ETSI approach to interoperability are the delivery of high quality interoperable standards, the availability of standardized test specifications and the practical provision of interoperability events.

With this in mind, experts from ETSI have contributed to the development and review of this present document (C2C-CC Demonstrator 2008). While the primary intention is that these demonstration scenarios are used in the Car 2 Car Forum and Demonstration 2008 it is hoped that the work will continue beyond that, i.e. the results will be valued input to standardization in this domain It is strongly believed that the availability of a complete set of well-defined standardized interoperability scenarios and associated specifications is an important factor in the delivery of truly interoperable car-to-car products and systems.

Introduction

The Car 2 Car Communication Consortium is actively driving harmonization, standardization, and market introduction of V2X systems. It is leveraging research results, incubating them towards maturity, and consolidating them for standardization since successfully standardized and harmonized systems are very attractive from an economic point of view, hence having best chances to get deployed on the mass market. Having said this it is essential to actively promote the harmonization of technologies and to enforce the adoption and validation of these. Furthermore, it is required to demonstrate the viability of such technologies.

The C2C-CC Demonstration 2008 intended to illustrate to the public that European vehicle manufacturers and suppliers jointly work on the elaboration of globally harmonized standards. The main objectives were to show interoperable applications across the different vehicle manufacturers and interoperable communications units from different suppliers working seamlessly together.

As a result of the C2C-CC Demonstration 2008, interoperability between the applications of nine vehicle manufacturers based on different communications equipment provided by five suppliers has been successfully demonstrated. Although the demonstrations did not include a complete set of functionalities required for a final system, it was shown for the first time that interoperability was not limited to the application level only but included all communications layers. Therefore, the technical specifications used for the C2C-CC Demonstration 2008 as well as the corresponding results thereof are considered valuable input for future ITS related standardization work.

The present document is a consolidated review of the C2C-CC Demonstration 2008 to which participants such as Alpine, Audi, BMW, Daimler, Delphi, Denso, DLR, ETSI Centre for Testing and Interoperability, Fraunhofer FOKUS, Fiat, Hitachi, Honda, NEC, Opel, Renault, Renesas, Siemens Austria, University of Applied Sciences Saarland (HTW), Volvo and VW actively contributed. The intention of the present document is to serve as basis for future standardization work.

1 Scope

The present document provides:

- a detailed description of Use Cases demonstrated during the C2C-CC Forum and Demonstration event 2008;
- a report on the corresponding technical specifications of the demonstrator system covering all communication layers.

It is understood that the C2C-CC demonstrator 2008 did not implement the complete functionality of a final system. However the implementations have been specified with a high demand with respect to interoperability and the current standardization status Therefore, it is intended that this report contributes to future standardization work:

- with a basic description of selected use cases;
- with a definition of a basic set of Application Support Facilities (CAM/DNM), which enable the selected use cases;
- with the specification of basic settings of the underlying communication layers (PHY-TRA) including communication security.

Moreover, the present document gives an example on how to visualize basic technological mechanisms such as Application Support Facilities (ASF), which are usually hidden to an end-user. This is can be of great value for future public demonstrations and interoperability tests.

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific.

- For a specific reference, subsequent revisions do not apply.
- Non-specific reference may be made only to a complete document or a part thereof and only in the following cases:
  - if it is accepted that it will be possible to use all future changes of the referenced document for the purposes of the referring document;
  - for informative references.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

2.1 Normative references

The following referenced documents are indispensable for the application of the present document. For dated references, only the edition cited applies. For non-specific references, the latest edition of the referenced document (including any amendments) applies.

Not applicable.

2.2 Informative references

The following referenced documents are not essential to the use of the present document but they assist the user with regard to a particular subject area. For non-specific references, the latest version of the referenced document (including any amendments) applies.

[i.1] C2C-CC Version 0.38: "C2C-CC Demonstrator 2008".

[i.2] ETSI TR 102 638 (V1.1.1): "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions".

[i.3] "CODAR VIEWER - A V2V COMMUNICATION AWARENESS DISPLAY". Matthias Kranz, Matthias Röckl, Andreas Franz, Thomas Strang, Conference Proceedings of "Pervasive 2008".

NOTE: Available at http://www.pervasive2008.org/Papers/LBR/lbr19.pdf.

[i.4] C2C-CC: "Single-Hop Broadcast Network Header", Version 0.2, C2C-CC NET WG, work in progress, February 2008.

[i.5] C2C-CC: "Network Header Structure and Common Network Header", Version 0.2, C2C-CC NET WG, work in progress, February 2008.

[i.6] C2C-CC: "Security Demonstration C2C-CC Demo 2008", C2C CC SEC WG, Part of the C2C CC Security Demo Pack.

NOTE: Available at https://www.car-2-car.org/groupware/mydms/op/op.Download.php?documentid=1177&version=1.

[i.7] C2C-CC: "Decentralized Environmental Message-v1.0", C2C-CC APP WG, February 2008.

[i.8] IEEE 1609: "Family of Standards for Wireless Access in Vehicular Environments (WAVE)".

[i.9] IEEE Draft Std P802.11p/D1.0, February 2006 , Part 11:wireless LAN Medium.

[i.10] IEEE P802.11p/D3.0, Part 11: Wireless LAN Medium Access Contrl (MAC) and Physical Layer (PHY) Specifications: Amendment: Wireless Access in Vehicular Environments (WAVE), Draft 3.0, July 2007.

[i.11] IEEE P802.11-2007: "Wireless local area networks".

[i.12] ETSI EN 302 571: "Intelligent Transport Systems (ITS); Radiocommunications equipment operating in the 5 855 MHz to 5 925 MHz frequency band; Harmonized EN covering the essential requirements of article 3.2 of the R&TTE Directive".

[i.13] ISO/TS 18234-4: "Traffic and Travel Information (TTI) -- TTI via Transport Protocol Expert Group (TPEG) data-streams -- Part 4: Road Traffic Message (RTM) application".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TR 102 638 [i.2] and the following:

application: computer program that defines and implements a use case functionality or a set of use cases functionalities

NOTE: A vehicular application is composed of software modules which receive data from vehicle sensors, communication networks and data bases, process them and deliver expected, consistent results to human users (via a HMI), generate communication or directly control the vehicle electronics.

Application Support Facilities: facilities which are common to several applications

NOTE: These application support facilities are regrouped into a Facilities layer which contains some generic application elements (middleware), presentation and session layers of the Open System Interconnection (OSI) Reference Model.

Use Cases: describes a vehicular customer's value adding service element provided by a local or global information system with a road safety, traffic efficiency, comfort or any other goal

NOTE: The customer can be the owner of the vehicle, its driver or one of its passengers. The use case can be providing information to or gathering information from a human being. It can also be receiving directly data from the vehicle electronics or any other embedded / remote electronic system and be controlling the vehicle electronics or any other embedded / remote electronic systems.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 102 638 [i.2] and the following apply:

ACC	Adaptive Cruise Control
API	Application Programming Interface
APP	APPlication layer
APP	Application Layer
ASCII	American Standard Code for Information Interchange
ASF	Application Support Facilities
BER	Basic Encoding Rules (ASN.1)
C2C	Car 2 Car
C2C-CC	Car 2 Car Communication Consortium
CAM	Co-operative Awareness Message
CCH	Control CHannel
DGPS	Differential GPS
DNM	Decentralized environmental Notification Message
EDCA	Enhanced Distributed Channel Access
EDCA	Enhanced Distributed Channel Access
EGNOS	European Geostationary Navigation Overlay Service
EPFL	Ecole Polytechnique Fédérale de Lausanne
ESP	Electronic Stability Control
EU	European Union
EV	Emergency Vehicle
FCS	Frame Check Sequence
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HMI	Human-Machine Interface
ITS	Intelligent Transport System
LLC	Logic Link Control
MAC	Medium Access Control
MC	MotorCycle
MSAS	Multi-Functional Satellite Augmentation System
NET	NETwork layer
OEM	Original Equipment Manufacturer

NOTE: In this context this refers to vehicle manufacturers.

OSGi	Open Services Gateway initiative
OSI	Open System Interconnection
OV	Oncoming Vehicle
PER	Packed Encoding Rules
PHY/MAC	Physical Layer / Medium Access Control Layer
PHY-TRA	Physical Layer – Transport Layer
QoS	Quality of Service
QoS	Quality of Service
RSU	Road Side Unit
SAE	Society of Automotive Engineers

SAP	Service Access Point
SEC	SECurity
SNAP	Sub-Network Access Protocol
STA	Station
STP	Simple Transport Protocol
TRA	TRAnsport layer
V2V	Vehicle to Vehicle
V2X	Vehicle to Vehicle or/and to road Infrastructure
VRU	Vulnerable Road User
WAAS	Wide Area Augmentation System
WAVE	Wireless Access in Vehicular Environments

4 Basic Application Support Facilities

The use cases described in the present document require the following basic functionalities:

- Periodic exchange of position and status information among the vehicles.
- Exchange of information related to a particular event.

In the context of the C2C-CC Demonstrator 2008 such basic, generic functionalities and related messages were summarized as "Basic Services". However, the present document follows the terminology currently used by ETSI TC ITS WG1 according to [i.2]. Accordingly, the Facilities layer provides generic functionalities that can be shared by several applications and their assigned use cases.

The Application Support Facility, which exactly provides the aforementioned required functionalities, is named "Messages Management". It is formatting common messages (CAMs, DNMs) out of the data of different applications, which are finally to be transmitted via the Networking and Transport layer. On the other hand, it also receives these common messages, and dispatches relevant data to requesting applications.

Co-operative Awareness Messages (CAM) are mainly used for permanently providing and evaluating essential data enabling vehicle co-operations and vehicle-road infrastructure co-operations. Decentralized Environmental Notification Messages (DNM) are used to distribute and to evaluate information related to road events.

This Application Support Facility must be able to generate common messages according to the rules defined for critical road safety applications and the network layer. CAMs are always communicated on the CCH (Control Channel positioned between 5,895 - 5,905 GHz within the allocated European ITS frequency band). DNMs will be using one of the ITS allocated channels. Whether to use the CCH or one of the other channels depends on the priority level of the information to be transmitted in the DNM (e.g. maximum latency time). On the receiver side this Application Support Facility must process received CAMs and DNMs in order to extract data for further use by local applications. The receiver must also be able to process DNM Cancellation messages. Moreover, the forwarding of DNMs must be stopped as soon as the defined message expiry time has been reached or a corresponding Cancellation DNM has been received indicating that this road hazard is no longer existing.

Another preconditioned Application Support Facility is Security Management. This instance enables the execution of security functions requested by the security layer (e.g. signing and verifying messages, verifying certificates, etc). These functions are indispensable for the system to defend against attacks (e.g. the injection of faked messages) and to preserve the driver's anonymity on sent messages. These capabilities are of superior importance with respect to safety requirements and customer acceptance.

5 Use Cases

This clause describes the use cases selected for the C2C-CC Demonstrator 2008 in a more descriptive way. The technical details to implement the corresponding applications are explained in annex A. The names of the use cases originally defined in the context of the C2C-CC Demonstration 2008 [i.1] have been adapted to the terminology currently used within ETSI TC ITS. Moreover, the corresponding use case identifiers have been applied. The base for this is [i.2].

5.1 Hazardous Location Notification / Stationary Vehicle Warning (Use Case Ident.: 00008/00004)

Stationary Vehicle Warning is a special Hazardous Location Notification use case where an immobilised vehicle has a problem, i.e. accident or breakdown. This hazard persists at the vehicle location for the time of the vehicle problem. Two Application Support Facilities are used to provide other vehicles information about the hazard: Co-operative Awareness Message and Decentralized Environmental Notification Message.

Co-operative Awareness Messages provide a fast direct communication link in the vicinity (one-hop communication range). It periodically broadcasts the basic vehicle status. The vehicle having the problem adds the information about its warning lights and, if applicable, information about existing vehicle problems or crash event to the Co-operative Awareness Message.

Decentralized Environmental Notification Messages allow vehicles to provide a more specific information within a wider area (multi-hop). A detection algorithm monitors the vehicle state. If the state indicates a vehicle problem that might be a potential danger to other road users a corresponding Decentralized Environmental Notification Message is sent (cause code `brokenDownVehicle(31)`). For the demonstration, a simplified hazard detection has been applied, which only verifies whether the vehicle is stationary with switched on emergency flashing lights. Depending on the duration of the problem, the vehicle updates the information after a while. If the problem is resolved a corresponding cancellation message is sent (also DNM).

The receiving vehicle decides if the information is relevant for the driver. This is done based on information of the first, the second, or a combination of both communication mechanisms, depending from the information policy of the receiving vehicle. If appropriate it gives a warning of the driver.

A receiving vehicle tries to match the information about the breakdown vehicle to its predicted path. This is done by matching the own position on the trace leading to the hazard location. The trace is obtained from the DNM sent out by the breakdown vehicle. The drivers of affected vehicles should get a timely warning before they reach the hazard location. The detailed HMI design is the responsibility of each OEM.

5.2 Roadwork Warning (Use Case Ident.: 00018)

The Roadwork Warning provides information about a construction zone to incoming vehicles in the immediate vicinity by sending Co-operative Awareness Messages. Furthermore more detailed information about the construction zone is provided by periodic sending of Decentralized Environmental Messages.

The receiving vehicle decides if the information is relevant for the driver. If appropriate a corresponding warning is issued. The drivers of affected vehicles should get a timely warning before they reach the hazardous spot.

5.3 Emergency Vehicle Warning (Use Case Ident.: 00001)

Emergency Vehicle Warning provides information to the driver about an approaching emergency vehicle. Based on the recurring Co-operative Awareness Message (CAM), the emergency vehicle will send information on the status of its siren and light bar. In the case that one of these two systems is active, the drivers of the receiving vehicles are warned. Additional information on the (relative) position and movement direction of the emergency vehicle can be conveyed and presented to the driver. Making way for the emergency vehicle will be improved, because of the early and precise information provided via the HMI. This is a crucial improvement compared to the conventional audio-visual warning via siren and light bar.

5.4 Motorcycle Warning (Use Case Ident.: 00020)

Looking into accident statistics, we can observe that the most common accident configurations - representing around 30 % of all accidents involving a motorcycle - are intersection accidents with perpendicular paths of the colliding vehicles and left turn accidents where the other vehicle cuts the path of the oncoming MotorCycle (MC).

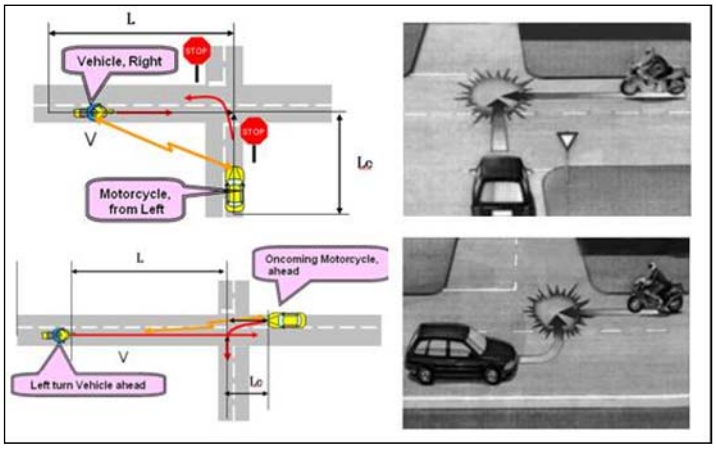


Figure 1: Accident Hotspots for Motorcycles

The other vehicle driver usually simply oversees the oncoming motorcycle or misinterprets its distance and speed. The Motorcycle Warning Use Case aims to address these accident types, which are mostly fatal for the motorcycle driver having the lowest protection. Bad motorcycle visibility is a major cause of MC accidents and a major challenge, which can partially be addressed by means of vehicle-to-vehicle communication systems.

A motorcycle as such is not a priority vehicle (like e.g.: emergency vehicle with activated sirens), but it is recognized at as a VRU (Vulnerable Road User) at EU level. The difference compared to other VRUs (like pedestrians and cyclists) is that MCs are mixing up in traffic together with other motorised road users at similar speeds. Therefore it is reasonable to explicitly identify MCs at communication level, in order to improve their protection based on adapted collision avoidance/mitigation strategies. The vehicle class is part of the CAM. Therefore motorcycles can be identified by simply evaluating this message.

It is important to notice that the motorcycle use case does not intend to provide the driver of an Oncoming Vehicle (OV) with information about the pure presence of a MC. Instead the goal is to effectively issue some warning in case of a safety critical situation where the involved vehicles are on a potential collision course and the time to collision is rather low.

The key initial situation for this use case is that the motorcycle has priority over the other oncoming vehicle. The MC approaches an intersection and intends to drive straight ahead. The OV, here the car, approaches the same intersection from another direction.

The OV is able to recognize the MC from the corresponding identifier of the CAM. When the MC is coming within a critically short distance relative to the OV, the system of the OV warns the driver. Similarly, the system of the MC also evaluates the information of the approaching OV (position, direction, speed), and if appropriate will generate a warning to the rider on the MC HMI. It is the responsibility of the OEMs how to provide this information to the vehicle driver in order to enable an appropriate reaction. Other vehicles in the communication range, but relatively far off the intersection, receive the same CAMs, but do not generate a warning, because there is no risk for collision.

6 Visualization of Application Support Facilities - Situation Monitor

A basic element of each V2X demonstration is to illustrate the connectivity among the single vehicles in order to enable the visitor to have a better understanding about the maturities of V2X. Such display visualizes technological functionalities like e.g. Application Support Facilities and Network-/Transport layer mechanisms, which are usually hidden to the end-user.

In order to visualize the basic technologies applied for the C2C-CC Demonstrator 2008, a so-called "Situation Monitor" has been defined. The Situation Monitor illustrates the current communication scenario across the entire demonstration area on a large screen. For this, a reasonable number of RSUs are connected to the Situation Monitor in order to overhear all exchanged messages. These RSUs are located at selected locations in order to cover the whole demonstration area.

The displayed information contains a real-time geographical overview about the road traffic situation, and may also be used to have a more detailed look on the content of messages exchanged between vehicles and RSUs.

The position of the single vehicles is obtained from the corresponding CAMs. Moreover, the Situation Monitor displays all road hazards on the map evaluating the corresponding DNMs.

Besides "Messages Management" (CAM, DNM), the Situation Monitor also visualizes another Application Support Facility: "Security Management". The implemented mechanisms are able to detect faked or unauthorized messages. For the demonstration, a dedicated communication system is located on the demonstration area, which continuously transmits "faked" messages. These messages are generated e.g. by signing with an invalid certificate. This is observed by the receiving vehicles and therefore the corresponding information is neither displayed nor evaluated. Instead, on the Situation Monitor these messages are highlighted as an attack in order to prove that the system can identify and avert such attacks.

The Situation Monitor of the C2C-CC Demonstrator 2008 was based on "CODAR" technology [i.3].

7 Lessons Learned

This clause aims to conserve the technological experiences which were gained from the C2C-CC Demonstrator 2008. This shall help to ease the preparation of future demonstrations and to identify potential technical challenges for the standardization of Applications and Application Support Facilities.

It is commonly known that the accuracy of positioning systems plays an important role for co-operative applications, particularly for collision avoidance. There are different approaches to tackle related problems. There are technological solutions to improve positioning like e.g. differential GPS (DGPS). Usually, a terrestrial transmitter sends additional data, which can be fused with the satellite data in order to calculate an improved position. Some implementations cause operative costs, because the correction data has e.g. been received via cellular communication devices. But there are also solutions where the data required for the correction of the position of a Global Navigation Satellite System (GNSS), such as GPS or Galileo, is sent out via geostationary satellites (WAAS, EGNOS, MSAS). Accordingly, DGPS seems to be a potential candidate to achieve the required positioning accuracy. For nodes, which do not support DGPS it is proposed to not only transmit the calculated position but also the raw GPS data in order to calculate a corrected position on the receiver side. This might be of particular interest for the interaction between vehicles and RSUs. RSUs might integrate DGPS functions or even provide correction data. However, related mechanisms appear relatively complex and consume additional bandwidth. For some co-operative applications related to collision avoidance the exact absolute position is not relevant. Only the relative positions of the affected vehicles are of interest. This is sufficient for calculating a potential collision point and the estimated time to collision. There are algorithms, which are relatively straight forward, but which are not easily to implement. Such algorithms are based on the assumption that the calculated positions of all nodes located in the same area at the same time have the same displacement vector caused by the same atmospheric interferences. Therefore, the relative positions of the involved vehicles (distance, and orientation) are very accurate. But in practice, it turns out that the different vehicles will be equipped with positioning devices of different quality. This means that maybe one vehicle still receives the weak signals from a particular satellite but the neighbour does not receive it anymore. Correspondingly, the relative position of the vehicles becomes increasingly inaccurate. Therefore applications based on relative positioning need to consider the existence of different positioning receivers in the single vehicles.

Network congestion is also a subject, which is very well known. So far the continuous sending of co-operative awareness information has not been discussed as a critical topic with respect to channel load. This is right for the case that the corresponding message is only send out every second by each node. But during the preparation for the C2C-CC Demonstrator 2008, it turned out that some co-operative collision avoidance applications require a higher transmission rate. Transmission rates of up to 10 Hz have been considered as reasonable for the intersection scenario. In this context it is worth mentioning that this makes only sense in case the involved nodes provide accurate position updates also at a rate of 10 Hz. However, if CAMs are sent at such high transmission rate this scenario has to be considered in the development of congestion control measures. During the preparations for the C2C-CC Demonstrator 2008 it has been discussed that the basic repetition rate for CAM is set to 1 Hz. Only in case two or more vehicles detect a potential risk for a collision the repetition rate of these particular vehicles is increased. The limitation of the transmission power may help to decrease the network load. But this has to be defined in conjunction with the requirements of the application.

Finally also another topic is illuminated, which was only less discussed so far. The consumption of resources should be considered in the future specification of applications and application support facilities. During the implementation phase of the C2C-CC Demonstrator 2008, it turned out that the encoding and decoding of information requires remarkable system resources (processing time). The same is the case for encapsulation- and de-capsulation mechanisms, which guarantee message integrity, consistency and communication security. One clearly has to consider that the implementations were only at demonstrator stage and that a final software will be much more integrated and optimized. On the other hand, for the C2C-CC Demonstrator 2008, the functionalities were distributed on several hosts, each with remarkable performance. Instead the performance of a final, ready for marked, integrated solution will most probably be comparably limited; particularly in case a cheap system is desirable, which is also affordable for low-class vehicles. Therefore, future processes for the specification of data modelling and communication security should consider the system resources, which are related to the proposed solutions.

8 Final Remarks

This clause aims to help to obtain a very basic overview about important functionalities, which were not implemented for the C2C-CC Demonstrator 2008, but which are considered as indispensable for the proper operation of a final system. Many of these functionalities will most probably be presented in future V2X demonstrations. In this context it is worth mentioning that the efforts required for the visualization of technological functionalities will drastically increase, which will finally be an aspect to be considered for future demonstrations. However this does not influence the necessity of these functions for the real system.

Use Cases based on Decentralized Environmental Notification Messages have complex requirements with respect to the applied strategy for message dissemination. On one hand the network may not get congested in dense traffic scenarios. This should be tackled by underlying routing- and multi-hop forwarding mechanisms of the network layer and related congestion control measures. Transmit Power Control mechanisms are also aiming to reduce the load in dense network scenarios. On the other hand, a critical safety message should not get lost in sparse network scenarios, e.g. a rural road at night. Therefore, it is essential that DNMs are stored and "physically" transported in the relevant area (dissemination area) for the time of their validity. This is even truer for the market introduction phase where only a small amount of vehicles will be equipped with V2X communication devices. Following a reasonable coexistence of store and forward ("physical transport of queued messages") and multi-hop routing mechanisms is desirable.

Moreover the systems need to deal with DNMs related to the same event received from different sources. As a consequence the logical combination of these messages becomes a crucial part of the future system. In this context "message clusters" can help to ensure that received information can consistently be proceed. Beyond this logical combination can help to improve the reliability of received information. Thereby intrinsic security on application layer level is achieved, which could e.g. be used in conjunction with the Security Management Application Support Facility.

As already stated in clause 4, communication security is of high importance with respect to the requirements of targeted safety applications and for the customer's acceptance. Besides the identification of system attacks the anonymity of the driver is an important aspect. Corresponding functions of the Security Management Application Support Facility need to ensure this.

Up to now RSUs played only a minor role acting as some sort of "repeater" for received or predefined messages. Although this is an important aspect, particularly for the market introduction period, RSUs offer far more potential. In an intersection scenario just one single RSU may simultaneously enable e.g.:

- Safety applications aiming at collision avoidance.
- Provision of traffic information to the incoming vehicles.
- Collection of traffic information (e.g. travel times).

However future RSU systems will probably not only transmit information to incoming vehicles but also process data received from the vehicles. For the latter case RSUs may host different applications. Therefore some basic functionality is required, which enables RSUs to "advertise" the supported applications to the vehicles within their radio range. Only in this way these vehicles are aware about the required information to be transmitted to this particular RSU. A similar kind of "Application Announcement" has been already been described in the drafted WAVE standards (IEEE 1609 [i.8]). There the functionality is based on so-called "WAVE Announcement Action Frames". The basic approach of "Application Announcement" has to be adapted to the European frequency usage and related mechanisms for channel access. Such functionality may be e.g. considered in the definition of a particular "Co-operative Awareness Infrastructure Message", which may become part of the Messages Management Application Support Facility for RSU systems.

In this context it is worth mentionin that the communication system of RSUs and OBUs is basically the same. This is particularly true for the mechanisms of the Network & Transport- (including PHY/MAC) as well as of the Security layer. The required adaptations just concern small parts of the facility layer. In case a RSU is connected to a communication backbone (fixed network), e.g. in order to interact with a traffic management centre, appropriate applications on the RSU need to ensure that the information to be transmitted to the incoming vehicles is pre-processed according to the needs of V2X communication.

Annex A:
Technical Specification of required Applications, Application Support Facilities and Network-/Transport Layer Settings

This annex specifies all technical settings of the communication system, e.g. PHY- and MAC-parameters as well as header formats and protocol definitions including data models. It represents the full technical specification, which was used for the integration of the applications and communication platforms into the different vehicle environments of the C2C-CC Demonstrator 2008.

A.1 Physical Layer (PHY)

The PHY layer of the communication system is based on IEEE P802.11p/D3.0 [i.10]. This clause only fixes the parameters to be used and, optionally, values that differ from this standard draft, **when necessary for interoperability**.

Table A.1: PHY Parameters

Parameter	Value	Remarks
Centre Frequency	5,890 GHz (CH 178)	Optional 5,9 GHz. The choice of the channel depends on the results of interoperability and propagation tests.
Channel Bandwidth	10 MHz	
Transmit Power	20 dBm	Recommended value. Subject to changes after the propagation tests.
Default Rate	3 Mb/s	Provides the most robust modulation scheme. Higher rates could be used according to the results of the propagation tests.

A.1.1 Modulation

The applied modulation follows the specifications of IEEE P802.11-2007 [i.11]. Amendments of IEEE P802.11p/D3.0 [i.10] regarding operating symbol clock frequency tolerance (see clause 17.3.9.5), receiver minimum input sensitivity (see clause 17.3.10.1), adjacent channel rejection (see clause 17.3.10.2) and nonadjacent channel rejection (see clause 17.3.10.3) are not mandatory.

A.1.2 Frame Format

The frame format follows the specifications of IEEE P802.11-2007 [i.11] (no amendments in IEEE P802.11p/D3.0 [i.10]).

A.1.3 Transmit Power

According to EN 302 571 [i.12] RF output power limit (e.i.r.p.) is 33 dBm and the power spectral density limit (e.i.r.p.) is 23 dBm/MHz. Because of the limited number of vehicles for the demonstration a congested medium is not expected. Prior goal for the demonstration is a stable communication under the conditions given at the test track. Therefore the following rules a proposed:

- The recommended output power is 20 dBm.
- The output power shall not be higher than 30 dBm.
- No power adaptation is used.

A.2 Medium Access Control (MAC)

The MAC layer of the communication system is based on IEEE P802.11p/D3.0 [i.10]. This clause only fixes the parameters to be used and, optionally, values that differ from this standard draft, **when necessary for interoperability**.

A.2.1 Basic Medium Access

The basic medium access is implemented according to IEEE P802.11p/D3.0 [i.10].

The recommended value for aSlotTime is 13 µs (IEEE P802.11-2007 [i.11]). Optionally, a value of 16 µs is also allowed (IEEE P802.11p/D1.0 [i.9]). Considering the expected low channel utilization in the demonstration, STAs adopting different aSlotTime are expected to have the same probability of accessing the channel.

A.2.2 WAVE Mode

WAVE BSS mode is not used. Instead, only WAVE mode with the predefined BSSID "FF:FF:FF:FF:FF:FF" is used.

A.2.3 EDCA

Full Enhanced Distributed Channel Access (EDCA) support for WAVE prioritized access operations as defined in IEEE P802.11p/D3.0 [i.10], Clause 7.3.2.29 is not used. Instead, only the lowest priority Access Category (ACI 00) is used for every MAC data frame.

A.2.4 MAC Frame format

Only data frames of subtype 0000 and 1000 (Data and QoS Data) are used. STAs must be able to process both subtypes. The maximum allowed payload size is 1 500 Bytes.

A.2.5 Source MAC Addresses

Each supplier shall use the assigned MAC address space.

Option: The support of varying MAC addresses for security or privacy is optional. The trigger for varying the addresses will be set externally. The MAC layer has to guarantee the correct functionality of the communication, therefore the execution of the request to modify the address might be delayed. No algorithm is given for creating the new address.

A.2.6 Destination MAC Addresses

Only broadcast frames are used. Therefore, the destination MAC address must be set to FF:FF:FF:FF:FF:FF.

A.3 Network Layer (NET)

For the C2C-CC Demonstrator 2008, the Single-Hop Broadcast Network Header as specified in [i.5] is used.

A.3.1 Network Header Structure

The following text is taken from [i.4] and copied here for better understanding:

The C2C-CC Network Header has a variable size and format. It is comprised of two parts (see figure A.1):

- the C2C-CC Common Network Header is a subset of the C2C-CC Network Header present in every C2C-CC data packet. The C2C-CC Common Network Header has a fixed size and format;

- the C2C-CC Extended Network Header is an optional subset of the C2C-CC Network Header added after the C2C-CC Common Network Header. Size and format of the C2C-CC Extended Network Header are variable.

Only the C2C-CC Common Network Header is used in the C2C-CC Demonstration 2008.

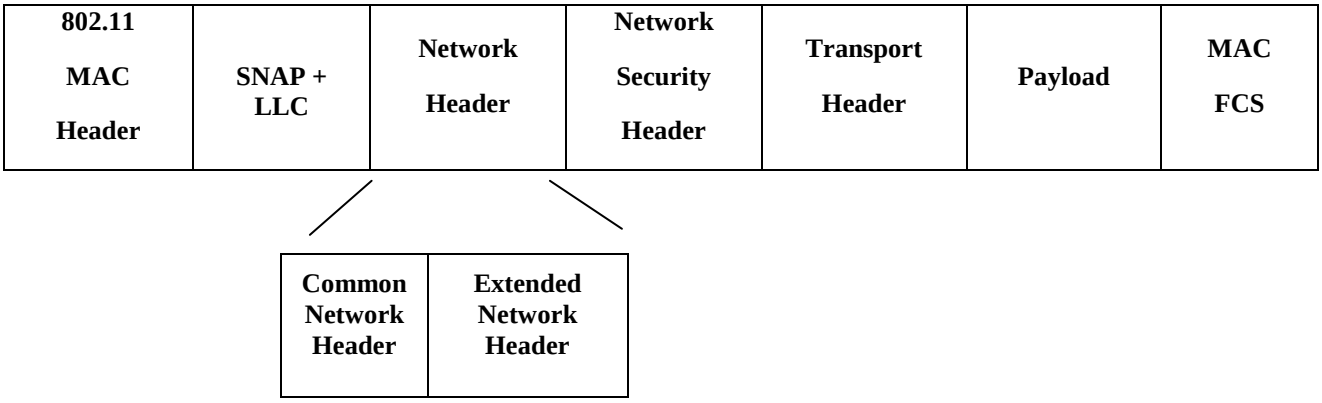


Figure A.1: Structure of the C2C-CC Network Header

A.3.2 Common Network Header

The C2C-CC Common Network Header is described in [i.5].

A.3.3 Common Network Header Format for Single-hop Broadcast

The Single-Hop Broadcast Network Header is specified in [i.4] and its ASCII representation is re-called in figure A.2, including some parameter settings.

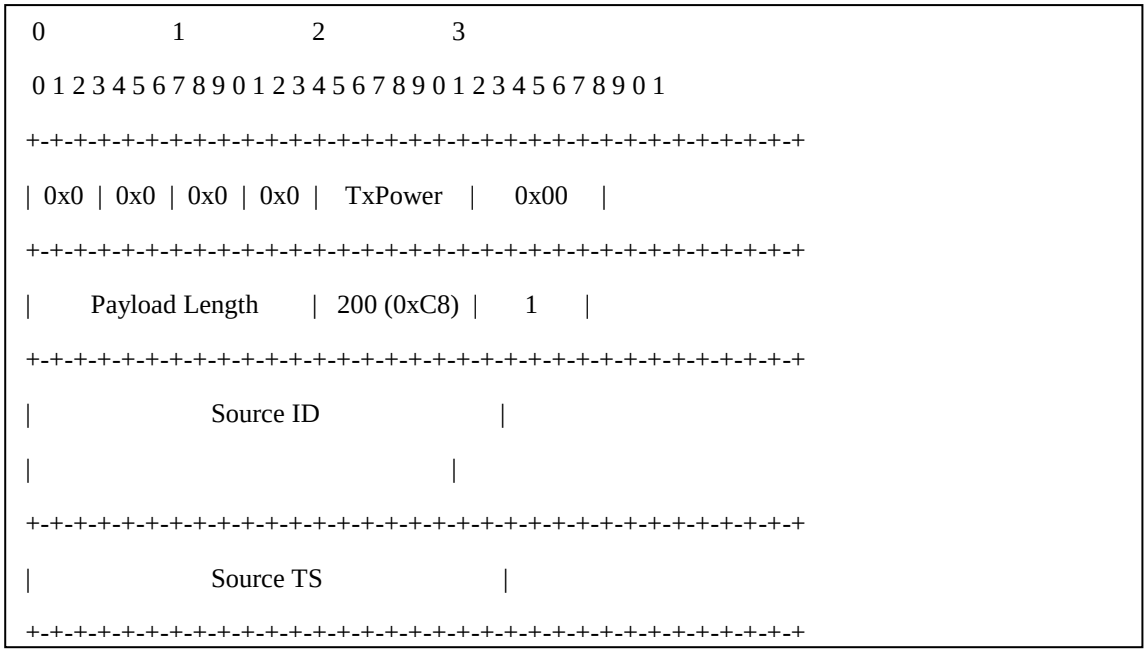


Figure A.2: ASCII Representation of the Single-hop Broadcast

The Ethernet type for the C2C-CC NET Header is 0x0707.

A.3.4 Network Security Header

The network security header will not be used for the C2C-CC Demonstrator 2008. Instead security is implemented on application layer, see clause A.5.

A.4 Transport Layer (TRA)

A Simple Transport Protocol (STP) header is used to differentiate the data traffic of different applications. In order to identify STP on network layer level the field *Next Header* of the C2C-CC Common Network Header is set to 200.

The format of the STP header is shown in figure A.3 as ASCII representation in network byte order.

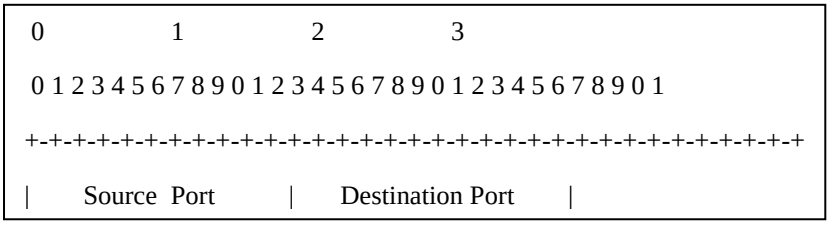


Figure A.3: ASCII Representation of the STP Header

`Source Port`
16-bit unsigned integer: Port of the sending application. If not used, the source port is zero.

`Destination Port`
16-bit unsigned integer: Port of the destination application.

A.4.1 Port Space

The STP port space, i.e. the set of values from which source and destination ports are chosen, is the same as for UDP. This means that well-known UDP ports should not be used. Please note that this limitation is only valid for the C2C-CC Demonstrator 2008.

It is recommended to use ports equal or higher than the default ports.

A.4.2 Default Ports

The **default destination port** for the C2C-CC Demonstrator 2008 is defined to **5000** (decimal). This default value should be used whenever applications do not define their own port.

Additionally, for the demo two destination ports for standard application data are defined:

5001: Co-operative Awareness data.

5002: Decentralized Environmental Notification Data.

Since no demo use case requires bi-directional communication, the source port is not relevant. Nevertheless, it is recommended to set the source port to the same value as the destination port.

A.5 Security (SEC)

Detailed information on the security daemon, its integration and mode of operation can be found in the Security Demo 2008 document [i.6].

For the purpose of the C2C-CC Demonstrator, security is integrated in the application layer by means of a function call in the Co-operative awareness application. The function calls are wrappers to abstract from calls to the security daemon using sockets. The main functions the daemon provides are:

- **Encapsulation:** A message, which is to be sent, can be signed with a specific key (e.g. private pseudonym key). The Security Header then includes the signature of the message and the corresponding certificate.
- **Decapsulation:** An incoming message with a security payload is passed to the security interface *verify* method in order to check its integrity, authenticity and trustworthiness => *verify(message, key, algorithm).*

For the integration of the security mechanisms, the encapsulation and decapsulation functions of the security module can be integrated using a simple demo API (refer to clause A.5.1). The Security Working Group maintains source code for:

- The security daemon: Software supporting encapsulation and decapsulation of messages.
- Wrappers: Functions for the inclusion in existing code to call the security daemon. Currently, there are wrappers for pure C, Java and OSGi.
- Tools for certificate and key generation

The following clauses outline the wrapper functions to be used for the demo. They are copied from the Security Demo 2008 document [i.6] for convenience.

A.5.1 Security Demo API

This clause briefly describes the APIs for integrating the described security module at any point in the protocol stack of a vehicular communication system. Based on the assumption of a signing/verification device or module, it provides byte-transparent signing and verification of messages based on generic formats for secured messages (the module currently supports WAVE 1609.2, a simple format by EPFL, and a dummy format with no security).

For the C2C-CC Demonstrator 2008 any management of certificates is handled transparent to the applications (and potentially even the communication system) using the daemon module.

The basic operations of the security modules are:

1) *Encapsulation* (Signature generation and certificate attachment).
2) Decapsulation (Signature verification and certificate interpretation/validation).

Encapsulation refers to attaching a certificate to a message and signing it. Decapsulation is the reverse process.

For the purpose of the C2C-CC Demonstrator 2008, it is assumed that all relevant information necessary for both processes are available in - and/or transparently managed by -the security modules. Hence for encapsulation it is sufficient to submit the bytes to be signed to the module. For decapsulation, i.e. reception processing there are two main aspects of security processing:

1) *Signature verification*, transparent to the application.
2) *Certificate verification*, with an output that may be relevant to the application.

The daemon indicates the security status of the packet. For this the *attackInfo* variable of the wrappers indicate whether the corresponding signature and the certificate could be verified.

The following clauses outline simple interfaces for adding security processing to applications in a transparent way. Corresponding C- and the Java /OSGi Interfaces are described. The interfaces are called "wrappers" or "frontends" and can directly be integrated into the application code.

A.5.1.1 Pure C Wrapper

The C wrapper is a pure C library for using the security daemon modules.

Example code: c-frontend/src/example.c file.

A.5.1.1.1 Configuration

By default, the *encap* and *decap* methods assume that the daemon runs on localhost and listens on the ports 4215 (signer) and 4216 (verifier). Those values are set in the *init_encap()* and *init_decap()* methods in the given config struct:

```
struct daemon_conf_t {
    char* addr;  /**The host where the daemon runs */
    int port;   /** The port the daemon listens to */
int32_t socketfd; /** The socket the daemon is
             connected to after initialization */
    int type;            /** Either encap or decap */
};
```

A.5.1.1.2 Encapsulation (Signature generation and certificate attachment)

The method takes the given input and submits it to the daemon. The demon encapsulates the given packet according to its specification. *Length* is adjusted to the actual length of the returned array of bytes. The method creates a newly allocated array of unsigned chars and does not change the given array.

```
unsigned char * encap(struct daemon_conf_t * conf, unsigned char * payload, int * length);
```

A.5.1.1.3 De-capsulation (Signature verification and certificate interpretation/validation)

Payload is a pointer to the data (an unsigned char array). Length will hold the size of the stripped off payload afterwards. The method creates a new array and does not change the given one. The *attacked field* is set to values indicating the verification result as given by the daemon. It holds its values in an unsigned char array that is assigned to the given *attackInfo* pointer.

```
unsigned char * decap(struct daemon_conf_t * conf, unsigned char * payload, int * length, unsigned
char * attackInfo)
```

A.5.1.2 Java/OSGi Wrapper

This interface is provided as JAVA classes. A corresponding .jar file can be included in the classpath for use in pure Java programs. Alternatively it can be loaded as a library bundle in an OSGi framework.

The interface definition for verifiers and signers - *IVerifier* and *ISigner* are provided separate from the implementations (*Verifier* and *Signer*). The *IVerifier* interface also defines values for interpreting the flags provided by the daemon.

Example code: java-wrapper/src/Example.java.

A.5.1.2.1 Configuration

By default, the *Signer* class assumes that the daemon runs on localhost and listens on the ports 4215 (signer) and 4216 (verifier).

A.5.1.2.2 Encapsulation (Signature generation and certificate attachment)

```
public class Signer {
public Signer()  {…}
/** Sign the given byte array. Length is part of array definition in Java. */
public byte[] sign (byte[] in){…}
```

A.5.1.2.3 De-capsulation (Signature verification and certificate interpretation/validation)

```
public class Verifier {
public Verifier(){}
/** Verify the given byte array. Length is part of array definition in Java. Attacked must be an
array of int in of size 1, where attacked[0] will hold the attacked value as defined for the Socket
interface.*/
public byte[] verify(byte[] in, byte[] attackInfo){…}
}
```

In addition, for convenience, the *Verifier* implementation contains a method for checking the *attackFlags*:

```
public Boolean isFlagSet(int flag, byte[] attackInfo)
```

The flag values are defined in the *IVerifier* interface and correspond to the checks for valid certificates and signatures.

A.5.1.3 Key Management

Correct operation of the daemon requires:

- Correctly installed root certificates for every node.
- Installed node certificates for every node.

These keys will be distributed before the C2C-CC Demonstrator 2008 by the security working group. For testing purposes, the ca-tool and key-gen tools can be used to create valid keys.

A.6 Specification of Required Application Support Facilities

A.6.1 Interface to Transport Layer (SAP)

All applications are respectively based on the following to Messages Management Support Facilities:

- **Co-operative Awareness**, destination port 5001, secured communication.
- **Decentralized Environmental Notification**, destination port 5002, no secured communication.

On Transport Layer these facilities can be distinguished by the *destination port* field defined in the STP header. For the C2C-CC Demonstrator 2008 the following processing rules apply. Please note that the security is completely handled on application layer. However this decision and the corresponding handling are not fixed for the C2C-CC yet.

A.6.1.1 Co-operative Awareness Data

For the communication of Co-operative Awareness data Security mechanisms have to be applied.

Sending: Co-operative Awareness data consists of a "CoopAwareness" data element as described in the following clause. For the C2C-CC Demonstrator 2008 the data is BER coded. Before the message is sent the application layer passes the encoded byte stream to the security module for encapsulation. The byte stream returned from the SEC-module containing the encapsulated application data is passed to the transport layer together with the destination port 5001.

Receiving: The application layer identifies data received on port 5001 as Co-operative Awareness data. The application layer passes the received byte stream to the security module for checking and decapsulation. Only data successfully checked by the security module is processed by the application layer. The returned unwrapped data is to be decoded by the Application layer according to BER as "CoopAwareness" data element.

Please note, that for the use of Co-operative Awareness data the application layer must access information available in the Network header, e.g. position data.

A.6.1.2 Decentralized Environmental Notification Data

For the Demonstrator, no security mechanisms are applied on the communication of Decentralized Environmental Notification data.

Sending: Decentralized Environmental Notification data consists of a "DecentralizedSituation" data element as described in the following clause. The data is BER coded. The resulting byte stream is passed to the Transport layer together with a destination port 5002.

Receiving: The application layer identifies data received on port 5002 as Decentralized Environmental Notification data. The byte stream received from the transport layer is decoded by the Application layer according to BER as "DecentralizedSituation" data element.

Please note, that the application layer forwards Decentralized Environmental Notification data according to clause 6.3.

A.6.2 Sending Co-operative Awareness Messages

Any communicating node shall periodically send Co-operative Awareness Messages.

For the demonstration, Co-operative Awareness Messages are not forwarded and the repetition rate is fixed to 1 Hz with a maximum latency of 500 ms. Note that these requirements are much less restrictive than specified for a final system. This allows the use for normal, standard GPS receivers with any additional algorithms for positioning. However, Filtering data using vehicle speed, heading, and yaw rate is recommended.

Motorcycles send Co-operative Awareness Messages with a fixed repetition rate of 5 Hz. An increased repetition rate is required to enable a timely warning in the receiving vehicles. For future motorcycle and intersection demo scenarios with significant speeds (including de- and accelerations and direction changes) of the participating vehicles an increased repetition rate for all vehicles (maybe using situation adaptive schemes) needs to be considered.

GNSS-Raw data are not mandatory for the C2C-CCC Demonstrator 2008.

The data model of the CAM is given below:

```
CoopAwareness ::= SEQUENCE {
    protocolVersion INTEGER (0..255),
    originator  NET_OHEAD  OPTIONAL,    -- not relevant for demo
    intendedRange   Range,  -- fixed to 250m
    messageBody     CHOICE {
        vehicleData [0] VehicleAwarenessData,
        rsuData [1] RSUAwarenessData,
        }
    }

VehicleAwarenessData ::= SEQUENCE {
    vehicleType VehicleType,
    length  VehicleLength,
    width   VehicleWidth,
    longitudinalAcceleration LongAcceleration,
    yawRate YawRate,
    accelerationControl AccelerationControl,
    exteriorLights  ExteriorLights,
    taggedList  SET SIZE(0..32) OF TaggedValue OPTIONAL
    }

RSUAwarenessData ::= SEQUENCE {
    rsuType RsuType,
    taggedList  SET SIZE(0..32) OF TaggedValue OPTIONAL
    }
```

The definition of the included complex data types (data elements) can be found in annex B.

Depending on the scenario, the tagged list shall contain the following information:

- sireneInUse and/or lightBarInUse, if the corresponding systems are activated.
- causeCode = 31 (brokenDownVehicle), if the corresponding decentralized Environmental message is sent out.

- `causeCode = 3 (roadWorks)`, for RSU road works.

A.6.3 Processing of Decentralized Environmental Notification Messages

Decentralized Environmental Messages have to be handled and distributed between RSUs and vehicles according to rules. Rules include e.g. expiry time, neighbour table changes, coverage range, etc. For the C2C-CC Demonstration 2008, two types of Decentralized Environmental Notification Messages were used, respectively for the use cases "Stationary Vehicle Warning" and "Roadwork Warning". This implies that only two Cause Codes are needed:

- 3: Roadwork Warning.
- 31: Stationary Vehicle Warning.

The action ID is fixed for the message used for the Roadwork Warning use case. Each vehicle generating a Stationary Vehicle Warning creates a random action ID and re-uses it for subsequent updates (*DataVersion*>0). For the cancellation message dataVersion=255.

Decentralized Environmental Notification Messages shall be supported by Store and forward functions implemented by vehicles. Vehicles shall e.g. be able to store the received Decentralized Environmental Notification Messages to be forwarded to the RSUs connected to the Situation Monitor. However, dedicated specifications for store and forward are currently not available. To simplify matters, every vehicle (including the generating one) periodically retransmits received Decentralized Environmental Notifications with 1 Hz for the time of validity and as long as the vehicle is within the dissemination area. This store and forward function is required in the whole demonstration site. A coverage range of 600 m is set to ensure this coverage. For the simplicity reason, the forwarding between vehicles is not required by the demonstrations. Transmission of Decentralized Environmental Notification Messages shall be started 3 s after detecting the corresponding use case.

For the dissemination area 600 m and for the expiry time 3 000 s are used as default values. The expiry time shall be automatically reloaded to its default value, if the situation persists after 70 % (i.e. 420 s for Broken down vehicle, 2 100 s for RSU road works) of the expiry time as long as no cancellation is received. Each reloading of the expiry time implies that the *dataVersion* is increased by 1. In case that the whole version range is used up (such as for the case of RSU road works), a new message shall be generated.

For the C2C-CC Demonstrator 2008 a trace is used for location referencing. The trace consists of several waypoints leading to the situation location. For the demonstration the number of waypoints is fixed to eight. Each moving node sets waypoints according to the following criteria:

- The node has driven more than 150 m since the last waypoint.
- The node has driven more than 15 m since the last waypoint and the product of this distance and the change of heading exceeds 750 [m degree].

Each node buffers the last eight (relevant) way points and adds those points to a Decentralized Environmental Message to be transmitted.

Road side units have to create the waypoints based on map data.

Decentralized Environmental Notification Messages consist of management container, event container and location container. For the demonstration, the message set and corresponding parameter setting is as below. The information adapted from [i.7].

```
    DecentralizedSituation ::= SEQUENCE {
        management  DecentralizedSituationManagement,
        situation   Situation,  -- container with situation description incl. type, severity
                            -- simplified for 2008 demo
        location    DecentralizedSituationLocation
        }

    DecentralizedSituationManagement::= SEQUENCE {
        protocolVersion INTEGER (0..255),
        actionID    ActionID,   -- Unique identifier of information about a situation from one
originator once randomly generated by brake down vehicle and RSU.
        dataVersion INTEGER(0..255),
                     -- Version of message indicating updates from the same originator;
```

```
                                -- 0 indicates the first generation of the message,
                                -- 255 is used for the cancellation message.
                                -- Each time the expiry time is reset, the dataVersion is increased
by 1.
        generationTime  TimeStamp,
                                -- time of vehicle generating the message after encountering the
start location of the hazard)
        expiryTime  TimeStamp,  -- time when the message shall be deleted from all databases and
sending queues.
                                -- For the demonstration, 600s is used for Broken down vehicle,
3000s is used for RSU road works.
        reliability INTEGER(0..100),
                                -- probability of hazard information to be true at mgt.
                                -- set to 90 for demonstration.

        isNegation  BOOLEAN     -- negates the existence of a situation type at the given location.
                                -- Not relevant for Demonstration. Allways false
        }

    DecentralizedSituationLocation::= SEQUENCE {
        situationLon    Longitude,
                                -- longitude of reference position for situation.
        situationLat    Latitude,
                                -- latitude of reference position for situation.
        situationAlt    Altitude,
                                -- altitude of reference position for situation.
        destinationArea CHOICE{
                                -- describes the destination area with points or distances values
            ellipse [0] EllipLocData,
                                -- when the required destination area is in the shape of ellipse,
                                -- not relevant for demonstration.
            circle  [1] CircLocData,
                                -- when the required destination area is in the shape of circle.
                                -- For the demonstration, 600m is set to cover the whole test track
area.
            rectangle   [2] RectLocData,
                                -- when the required destination area is in the shape of rectangle
                                -- not relevant for the demonstration.
            ...
            },

        locationRef CHOICE{
            trace   [0] SEQUENCE SIZE (0..16) OF ShortPosition2D,
                                -- list of waypoints leading to the situation position the first
            ...
            }
        }

    Situation ::= CauseCode -- Very simplified situation description for Demo only!!!
```

The definition of the included complex data types (data elements) can be found in annex B.

Additional to these definitions the `Sequence` of `ShortPosition2D` of the `locationRef` data element is used and calculated in the following way.

The data element `ShortLongitude` provides the last 16 bit of a **relative** longitude coordinate with a granularity of 1/8 µ° (WGS84 coordinate system). Together with a Longitude of a reference position in the vicinity the complete position can be reconstructed. The `situationLon` value derived from the `DecentralizedSituationLocation` should be used as the reference value for the calculation.

`ShortLongitude` = the last 16 bit of (absolute trace point longitude - `situationLon`).

For reconstruction the absolute coordinate of a waypoint, the following formula shall be used:

absolute trace point longitude = `situationLon` + `ShortLongitude`

The values shall be encoded using two's complement. In this case for 16 bit within the range of $-32\ 768_{(10)}$ to $+32\ 767_{(10)}$. Each unit has to ensure to use relative positions with offsets smaller than the given range.

The value of `ShortLatitude` is calculated analogous.

Trace point #1 contains the position of the most recent waypoint, closest to the situation location.

A.7 Application Layer (APP)

A.7.1 Specification of Use Case Implementations (Applications)

A.7.1.1 Generation of Stationary Vehicle Warning

Each vehicle shall be able to automatically generate a Stationary Vehicle Warning. A simplified detection algorithm shall be used for the C2C-CC Demonstrator 2008:

- The situation is detected, when the vehicle is stationary (speed < 1 m/s) and the emergency flasher is switched on. The end of a situation is detected immediately after the emergency flasher is turned off.
- The expiry time shall be set to 600 s after generation time. The message is updated (new generation time, new expiry time, increased data version) if the end of the situation has not been detected and the last update is 420 s old. When the end is detected a cancellation is send out: data version is set to 255. The expiry time is equal to the expiry time of the last update message.
- The Reliability is set to 90% for the C2C-CC Demonstrator 2008. The cause at the corresponding event data is set to 31 = *"broken down vehicle"*. Additional optional parameters are not set. The event data are put to a Decentralized Environmental Message. The format and the handling are described in clause 6.3.

A.7.1.2 Motorcycle Warning

For C2C-CC Demonstrator 2008 no digital map with node and link data is required (the available map is mainly for visualization purposes). As a consequence for this application the road configuration (geometry + traffic rules) needs to be estimated on basis of the vehicle trajectories.

The coordinates of intersection are calculated with estimated loci of vehicles. The locus of the vehicle is estimated using position data, velocity, direction vector, etc. The cross point of two loci is regarded as the intersection. Priority of roads can be judged by driver's behaviour.

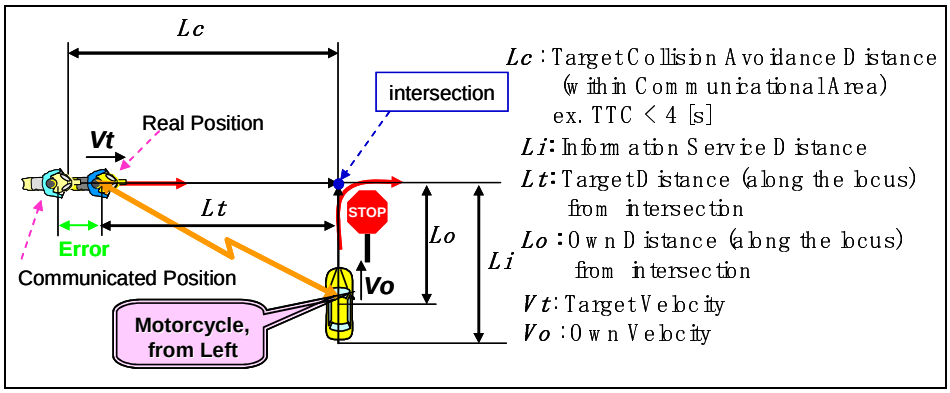


Figure A.4: Evaluation of potential collision courses

Because of GPS sensing delay and system delay, there is an error between the real position and the communicated position of the target vehicle. Before calculation of *Lt*, the coordinates of the target vehicle must be compensated (ex. GPS 1 s delay + System delay → 1,3≈1,5 [s] estimated).

State transition diagram for cars:

According to the scenario selected for the C2C-CC Demonstrator 2008 the car should stop at the stop sign and then - because the view is obstructed - roll forward slowly. In order to avoid a warning message every time a motorcycle is approaching the state diagram consists of 3 states (see below). In the "ordinary state" S0 (if the vehicle is not within the information service distance), no notification is given. Close to an intersection the system is in "information service stand-by" state S1. From this state an approaching motorcycle can trigger the notification for the car driver (S2).

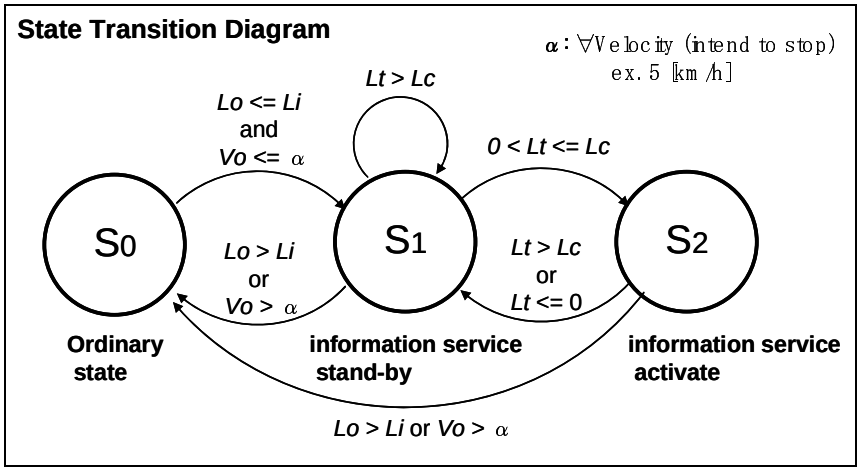


Figure A.5: State transition diagram for cars

If digital map with node and link information is available:

If a digital map is available the coordinates of the intersection are calculated with the node and link data. This simplifies the calculation of Lt and Lo. The state diagram remains unchanged.

A.7.1.3 Emergency Vehicle Warning

The Co-operative Awareness Message of the emergency vehicle shall follow these rules:

- The value of `VehicleType` shall be set to `emergencyVehicle`.
- The tagged list shall contain `SirenInUse` and `LightbarInUse` which can be set to `SimpleSystemState`, i.e. `unavailable`, `disabled`, `enabled`, `engaged`. However, as `engaged` and `enabled` mean the same here, only `engaged` shall be used. The sending part of the Emergency Vehicle Application will sign messages and attach a valid certificate. Therefore all receiving vehicles shall use the security daemon to verify the incoming messages.

When receiving a CAM it shall be checked whether it comes from an Emergency Vehicle (EV). This can be achieved by checking the value of the element `VehicleType` which in this case is set to `emergencyVehicle`.

In case an EV is approaching it shall be checked whether it is on emergency. This can be done by checking the two elements from the tagged list `SirenInUse` and `LightbarInUse`. If these data elements are set to `engaged` the corresponding emergency vehicle is on duty (`SimpleSystemState`). Please note that for the C2C-CC Demonstrator 2008 both values will be either set to `disabled` or `engaged` at the same time.

If the CAM reveals that the Emergency Vehicle (EV) is in emergency state, the following process should be executed:

- Check whether message is relevant (EV is following own car):
  - Check distance, speed and heading of the EV.
  - Ideally compare with own trace points to be able to decide whether EV is following.
- Predict the time of arrival of the EV at current position.
- If prediction says EV will reach own current position within 10 s the driver should be requested to stop the car at the roadside in order to let the EV pass.

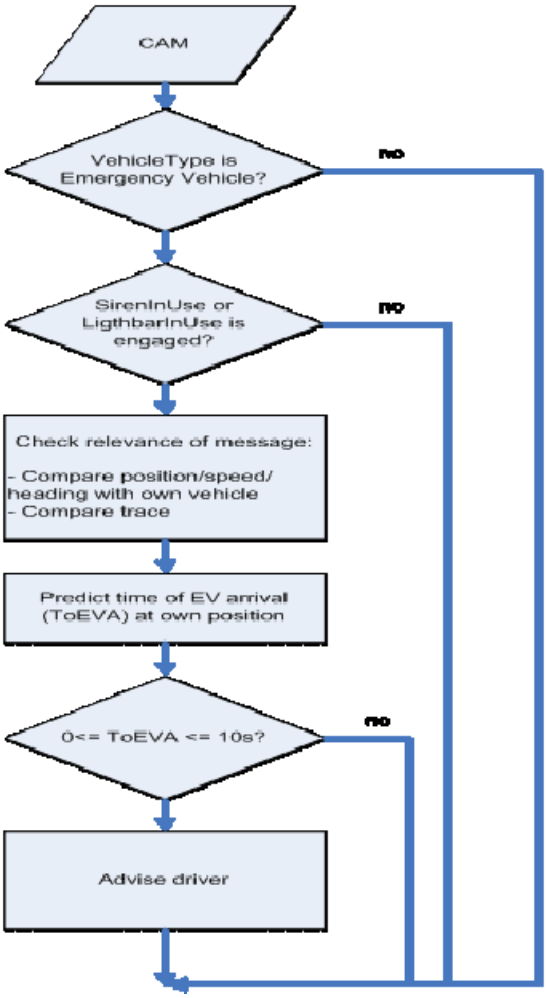


Figure A.6: Processing of Emergency Vehicle Information in CAM

A.7.1.4 Emergency Vehicle Warning

The Co-operative Awareness Message of the Roadworks-RSU shall follow these rules:

```
CoopAwareness ::= SEQUENCE {
    protocolVersion INTEGER (0..255),   -- fixed to 1
    intendedRange   Range,  -- fixed to 250m
    messageBody     CHOICE {
        rsuData [1] RSUAwarenessData,
        }
    }

RSUAwarenessData ::= SEQUENCE {
    rsuType RsuType,    -- fixed to movingWorkzone   (2)
    taggedList  SET SIZE(0..32) -- fixed to 1 entry
    }
```

The tagged list shall contain a cause-code which shall be set to `roadWorks(3)`.

"Decentralized Environmental Notification" allows the RSU to distribute more specific information within a wider area. The RSU sends out a corresponding *Decentralized Environmental Notification Message* with cause code `roadWorks(3)`. Depending on the duration of the roadworks situation the information is updated after a while. If the roadwork is finished a corresponding cancellation message is sent.

The Decentralized Environmental Notification Message for the use case Roadwork Warning shall follow these rules:

```
    DecentralizedSituation ::= SEQUENCE {
    management  DecentralizedSituationManagement,
    situation   Situation
    location    DecentralizedSituationLocation
    }

    DecentralizedSituationManagement::= SEQUENCE {
    protocolVersion INTEGER (0..255),
    actionID    ActionID,   --                      -- fixed to 3
        dataVersion INTEGER(0..255)
        generationTime  TimeStamp
        expiryTime  TimeStamp       -- 3000s is used here
        reliability INTEGER(0..100) -- fixed to 90
        isNegation  BOOLEAN     -- fixed to false
    }

    Situation ::= CauseCode                                     -- fixed to roadWorks (3)

    DecentralizedSituationLocation::= SEQUENCE {
    situationLon    Longitude, -- longitude of reference position for situation.
    situationLat    Latitude,  -- latitude of reference position for situation.
    situationAlt    Altitude,                           -- fixed to 0

    destinationArea CHOICE{
        circle  [1] CircLocData,        -- fixed to 600m.
    },

    locationRef CHOICE{
    trace   [0] SEQUENCE SIZE (0..16) OF ShortPosition2D,
                                        -- fixed to 8 waypoints
                                                leading to the situation position
    }
}
```

A.7.2 Warning Modules (HMI)

The algorithms for internal data processing and for HMI design are up to each OEM. Processing of the data includes combination of information, situational filtering of relevant information considering the current driving situations, and generation of driver information.

However, the basic functionality in each demonstrator is the same: E.g. for the Stationary Vehicle Warning the driver gets a corresponding notification about 10 s before he reaches the hazard location. Timing bases from the current speed, map data and the trace provided within the Decentralized Environmental Notification Message.4.

Annex B:
Data Element Definitions

B.1 General Type Definitions

This clause provides the definition of some general types.

B.1.1 AbsLatitude

Purpose	The data element provides an absolute latitude coordinate (WGS84 coordinate system) with a granularity of 1/8 µ°. The meaning of the described location is determined by the coordinate is determined by the context or by the definition of the derived data element.
ASN.1 Representation	`Latitude ::= INTEGER (-720000000..720000000)`
Notes	Compliant to SAE J2735.

B.1.2 AbsLongitude

Purpose	The data element provides an absolute longitude coordinate (WGS84 coordinate system) with a granularity of 1/8 µ°. The meaning of the described location is determined by the coordinate is determined by the context or by the definition of the derived data element.
ASN.1 Representation	`Longitude ::= INTEGER (-1440000000..1440000000)`
Notes	Compliant to SAE J2735.

B.1.3 ShortLatitude

Purpose	The data element provides the last 16 bit of a latitude coordinate with a granularity of 1/8 µ° (WGS84 coordinate system). Together with a Latitude of a reference position in the vicinity the complete value can be reconstructed. The meaning of the described location is determined by the context or by the definition of the derived data element.
ASN.1 Representation	`ShortLatitude ::= INTEGER (0..65535)`
Notes	Compliant to SAE J2735.

B.1.4 ShortPosition2D

Purpose	The sequence is used for instance for way point chains.

ASN.1 Representation	`ShortPosition2D ::= SEQUENCE {`
	`  shortLon  ShortLongitude,`
	`  shortLat  ShortLatitude`
	`  }`

Notes	Compliant to SAE J2735 DF_PositionShort.

B.1.5 Confidence

Purpose	The data element provides the symmetric interval of 95 % confidence level for a current reported value. If not defined differently the confidence limits of the interval are calculated based on the Granularity of the corresponding measurement data element according to: Limit = ± LSB_Value * 2 ^ `Confidence` 15 is set if no other value is available.
ASN.1 Representation	`Confidence::= INTEGER (0..15)`
Notes	The use of 95 % confidence intervals and the logarithmic scale is compliant to SAE J2735.

B.1.6 Range

Purpose	The data element provides a range with a granularity of 25 m.
ASN.1 Representation	`Range::= INTEGER (0..255)`
Notes	

B.1.7 SimpleSystemState

Purpose	Harmonized description of very general description of a system. This description is mainly used for optional systems at cars, e.g. ESP: `unavailable` not equipped or out of order. `disabled` switched of by user or due to driving situation, e.g. ACC below minimum speed. `enabled` switched on but no action, e.g. ESP in normal operation, limiter below limit speed. `engaged` switched on and in action, e.g. light bar flashing, limiter limiting speed. If `enabled` and `engaged` cannot be distinguished for a system `engaged` shall be used, e.g. light bar.

ASN.1 Representation	`SimpleSystemState ::= ENUMERATED {`
	`  unavailable (0),`
	`  diabled (1),`
	`  enabled (2),`
	`  engaged (3)`
	`  }`

Parameter range constraints	None
Notes	Similar enumeration is used at SAE J2735, however it is not harmonized within the standard.

B.2 AccelerationControl

Purpose	

ASN.1 Representation	`AccelerationControl ::= SEQUENCE {`
	`  brakePedal ENUMERATED { off (0), on (1) },`
	`  throttlePedalENUMERATED { off (0), on (1) },`
	`  cruiseControlENUMERATED { off (0), on (1) },`
	`  acc ENUMERATED { off (0), on (1) },`
	`  limiter ENUMERATED { off (0), on (1) }`
	`  }`

Notes	None

B.3 VehicleType

Purpose	The enumeration provides the categories of vehicle. In general this is depending on the form of the vehicle. If functional types apply to the vehicle, e.g. emergency vehicle, the corresponding type should be selected.

ASN.1 Representation	`VehicleType ::= ENUMERATED {`
	`  unknown (0),`
	`  car (1),`
	`  emergencyVehicle (6),`
	`  motorcycle (19),`
	`  ...`
	`  }`

Notes	See ISO/TS18234-4 [i.13] table rtm01.

B.4 LongAcceleration

Purpose	The data element represents the signed acceleration in direction of the element heading with a granularity of 0.01 m/s². Negative values indicate deceleration.
ASN.1 Representation	`LongAcceleration::= INTEGER (-2000..2000)`
Notes	Derived from SAE J2735 DE_Acceleration. Longitudinal acceleration is specified by the SAE standard based on the vehicles centre line. In order to get constant data we chose to base the definition on the Heading because this value is communicated between vehicles. In practice no difference has to be made based on the specification. According to the SAE Specification this data element can also be used for lateral acceleration. We specify the use of YawRate for turning movements only.

B.5 YawRate

Purpose	The data element represents the signed change of the element heading with a granularity of 0,01 deg/s.
ASN.1 Representation	`YawRate::= INTEGER (-2000..2000)`
Notes	Derived from SAE J2735 DE_YawRate. Not correctly defined at SAE J2735 Rev21.

B.6 NodeLatitude

Purpose	The data element describes the latitude of the reference point of a communicating node. The position of the sending antenna and at least one receiving antenna must be within 10 m distance of the reference position. The reference point is defined as follows: `Vehicle` middle of the vehicle's front. `Others` location of the sending antenna.
ASN.1 Representation	`NodeLatitude::= AbsLatitude`
Notes	None

B.7 NodeLongitude

Purpose	The data element describes the longitude of the reference point of a communicating node. The position of the sending antenna and at least one receiving antenna must be within 10 m distance of the reference position. For the definition of the reference see `NodeLatitude`.
ASN.1 Representation	`NodeLongitude::= AbsLongitude`
Notes	None

B.8 PositionConfidence

Purpose	The data element describes the 95 % confidence interval of the horizontal position. It is valid for the corresponding longitude as well as latitude measurement. The limits are based on the granularity of the latitude data element (approx.: 0,014 m). Limit = ± 0,01 * 2 ^ `Confidence`.
ASN.1 Representation	`PositionConfidence::=INTEGER(0..15)`
Notes	Similar to SAE J2735 Position Confidence.

B.9 LightbarInUse

Purpose	The data element describes the status of any sort of additional visible lighting-alerting system. For example, these additional visible lighting-alerting systems might be part of an emergency vehicle, transportation response vehicle, or maintenance vehicles.
ASN.1 Representation	`LightbarInUse::= SimpleSystemState`
Notes	Derived from SAE J2735. Enumeration adapted to the general `SimpleSystemState`.

B.10 SireneInUse

Purpose	The data element describes the status of any sort of audible alarm system beside the horn. This includes various common sirens as well as backup up beepers and other slow speed manoeuvring alerts.
ASN.1 Representation	`SireneInUse::= SimpleSystemState`
Notes	Derived from SAE J2735. Enumeration adapted to the general `SimpleSystemState.`

B.11 VehicleWidth

Purpose	The data element describes the vehicles' width at the widest point with a granularity of 0,01 m.
ASN.1 Representation	`VehicleWidth::= INTEGER(0..1023)`
Notes	Taken from SAE J2735 DE_VehicleWidth.

B.12 VehicleLength

Purpose	The data element describes the vehicles' length at the longest point with a granularity of 0,01 m. This shall include a trailer if present. If the length with trailer cannot be determined the maximum allowed length should be given and corresponding confidence value has to be provided.
ASN.1 Representation	`VehicleLength::= INTEGER(0..16383)`
Notes	Taken from SAE J2735 DE_VehicleLength.

B.13 ExteriorLights

Purpose	The bitfield describes the status of the most important exterior lights. The fogLightOn indicates the status of the tail fog lamp. If a vehicle is not equipped with a certain light the value is set 0. If one, more, or all lamps corresponding to a certain "light group" (e.g. front, back and side lamp of indicator) are not functional the corresponding bit is set if the light is switched on by the driver or automatically by a vehicle system. The turn signal and hazard signal bits should provide the corresponding switch status not the lamp status, i.e. they should not alternate with the blinking interval.

ASN.1 Representation	`ExteriorLights ::= BIT STRING {`
	`  allLightsOff (0),`
	`  lowBeamHeadlightsO (1),`
	`  highBeamHeadlightsOn (2),`
	`  leftTurnSignalOn (4),`
	`  rightTurnSignalOn (8),`
	`  hazardSignalOn (12),`
	`  automaticLightControlOn (16),`
	`  daytimeRunningLightsOn (32),`
	`  fogLightOn (64),`
	`  parkingLightsOn (128)`
	`}`

Notes	Taken from SAE J2735 with some more detailed specification on the data use.

B.14 CauseCode

Purpose	The enumeration indicates the cause of a potentially traffic relevant situation. (selected items only).

ASN.1 Representation	`CauseCode ::= ENUMERATED {`
	`  unknown (0),`
	`  accident (2),`
	`  roadWorks (3),`
	`  slipperyRoad (11),`
	`  brokenDownVehicle (31),`
	`  …`
	`}`

Notes	Taken from TPEG-TEC draft.

B.15 TaggedValue

Purpose	The data element represents an entry of a tagged list.

ASN.1 Representation	TaggedValue CHOICE {
	-- SAE compliant tags
	longAcceleration [2] LongAcceleration,
	exteriorLights [29] ExteriorLights,
	lightBarInUse[33] LightBarInUse,
	sireneInUse [50] SireneInUse,
	vehicleLength[70] VehicleLength,
	vehicleWidth [76] VehicleWidth,
	yawRate [81] YawRate,
	-- not SAE compliant values and tags
	causeCode [128] CauseCode,
	brakingActivity [129] BrakingActivity,
	brakingDetails [130] BrakingActivityDetails,
	rawGNSSdata [131] RawGNSSdata,
	…
	}

Notes	Taken from SAE J2735 with some addition to the used data elements.

B.16 WayPointList

Purpose	The data element represents a list of way points representing a linear geometry in longitude and latitude.

ASN.1 Representation	WayPointList SEQUENCE SIZE (0..16) OF {
	wpLongitude ShortLongitude,
	wpLatitude ShortLatitude
	}

Notes	Not compliant with SAE J2735. for demonstration only!

Annex C: Encoding

Encoding rules specify the binary representation of messages and information elements.

For the definition of the application data model ASN.1 was used. The defined messages shall be encoded according to Basic Encoding Rules (BER). For a later system Packed Encoding Rules (PER) should be used in order to safe bandwidth.

History

Document history		
V1.1.1	June 2009	Publication