UML Model of a Gateway for the Interconnection of IEEE 1609 and Controller Area Network

Silviana Juárez Chalini[1], Miguel Angel León Chávez[2], and Gladys Diaz[3]
[1]Universidad de la Cañada, [2]Computer Science Faculty, BUAP,
[3]L2TI – Institut Galilée – Université Paris 13 - Sorbonne

In the context of future vehicular applications, new models must to be defined to enable the access and interconnection of existing control systems. This paper presents a model of a Gateway for the interconnection between two networks, on one hand IEEE 1609, and on the other hand Controller Area Network (CAN). The former defines Wireless Access in Vehicular Environment (WAVE); the second is a serial communication bus that is used by the automotive industry for interconnecting all kind of devices into a car, such as ABS (antilock-braking system), etc. A possible application of the Gateway is to use the potential collision messages not only as warning messages for the vehicle driver but as control messages for the device controlling the brake into a vehicle. The Gateway's model is developed using the Unified Software Development Process and its language the Unified Modeling Language (UML).

I. Introduction

Wireless networks are been deployed to meet the requirements of many applications, such as those of the automotive. IEEE has specified a family of standards named Wireless Access in Vehicular Environment (WAVE) that defines the communications vehicle-to-vehicle and vehicle-to-infrastructure by means of the services provided by the following three layers: lower layers (physical layer using IEEE 802.11p, and MAC sublayer defined by IEEE 1609.4); network and transport layers (defined by IEEE 1609.3 services); and upper layers (initially defined by IEEE 1609.1; today it is a withdrawn standard). On top of these layers non-safety and safety applications can exist. WAVE specifies also security services for applications and management messages by means of IEEE 1609.2.

There are two types of WAVE devices for vehicular environments, the first is called roadside unit (RSU), it is stationary, and it is usually fixed along the road; the second type, known as the onboard unit (OBU), is usually mounted in the mobile vehicles.

WAVE provides user services ranging from comfort, e.g. wireless Internet access, up to information services to keep the driver informed about the road conditions, e.g. traffic flow, potential collisions to other vehicles, etc.

On the other hand, modern vehicles interconnect all kind of devices by means of at least two networks: critical and comfort. The former interconnects important devices such as antilock-braking system (ABS), engine-control management (ECM), electronic transmission control (ETC), steering-angle sensor (SAS), throttle-control management (TCM), and central electronic module (CEM).

The comfort network interconnects devices such as audio module, climate-control module, driver-door module, passenger-door module, power-seat module, rear-electronics module, road-traffic information, safety-restraint system, steering-wheel module, and upper electronic module.

These networks can be implemented using Controller Area Network (CAN) that is a serial communication bus supporting distributed real-time control with a high level of error control.

Basically, CAN defines two OSI layers: physical and data link layer. The upper layers are not standardized, however there are several commercial proposals at the application layer.

This paper proposes a gateway, at the application layer, allowing the communication between a CAN node and IEEE 1609 OBU. A possible application of the gateway is to perform read and write operations between both networks for using the potential collision messages not only as warning messages for the vehicle driver but as control messages for the device controlling the brake into a vehicle. The variables and their values, as well as their processing to generate the potential collisions message are out of the scope of this paper.
The paper presents a model of the gateway using the Unified Software Development Process and its language the Unified Modeling Language (UML). The rest of the paper is organized as follows, section II presents CAN and its services; section III describes the family of IEEE 1609 standards. Section IV presents the analysis and design models of the gateway. Conclusions and future research work are drawn in section V.

II. CAN Architecture

According to the ISO/OSI reference model, CAN [1, 2] defines the physical and data link layer, the last one divided in both Logical Link Control (LLC) sublayer and Medium Access Control (MAC) sublayer, version A of this standard named them object layer and transfer layer. Upper layers are not defined however there are several proposals such as CAL, CANopen, TT-CAN, DeviceNet, SDS, and CANKingdom. The services provided by each layer are briefly described next.

A. Physical Layer

It defines signals transmission and therefore deals with the description of bit timing, bit encoding, and synchronization.

B. MAC sublayer

It is responsible for message framing, arbitration, acknowledgement, error detection, and signaling. MAC sublayer is supervised by a management entity called Fault Confinement that is a self-checking mechanism for distinguishing short disturbances from permanent failures.
The nodes in a CAN network do not use any information about the network configuration, i.e. node addresses. An identifier names the content of a message. The identifier does not indicate the destination of the message but describes the meaning of the data, so that all nodes in the network are able to decide by message filtering whether the data is processed by them or not.
Therefore [1], whenever the bus is free, any node may start to transmit a message. If two or more nodes start transmitting messages at the same time, the bus access conflict is resolved by bitwise arbitration using the identifier. The mechanism of arbitration guarantees that neither information nor time is lost. During arbitration every transmitter compares the level of the bit transmitted with the level that is monitored on the bus. If these levels are equal the node may continue to send. When a “recessive” level is sent and a “dominant” level is monitored, the node has lost the arbitration and must withdraw without sending one more bit.

C. LLC sublayer

It is concerned with message filtering, overload notification, and recovery management. A CAN node sends a data frame through the LLC sublayer; this transaction must be single, self-contained operation independent of previous frame transactions. The transmitter maybe needs a delay in the transmission or waits for a LLC remote frame; the LLC is capable to generate at most two MAC overload frames to produce the delay. If the transmission fails the LLC sublayer can retransmit the frames that have lost the arbitration or have been disturbed by errors during transmission. On the other hand when a CAN node is a receiver, it may decide by frame acceptance filtering whether the frame is relevant or not, according to the information from the identifier.

III. WAVE Architecture

A WAVE device is composed by a stack of standards according to the ISO/OSI reference model, as Fig. 1 shows, these standards and their services are as follows:

A. IEEE 802.11p

IEEE 802.11p [3] is an extension of the IEEE 802.11. It describes the functions and services required by stations to operate in a rapidly varying environment and to exchange messages without joining a basic service set (BSS), i.e., without authentication, deauthentication and privacy services at the MAC sublayer, such as they are specified by IEEE 802.11.

B. IEEE 1609.4

The services provided by IEEE 1609.4 [4] at the MAC sublayer, are the following: channel coordination, channel routing, user priority, and MSDU data transfer; as they are briefly explained below:

Channel coordination: it coordinates the channel intervals according to the channel synchronization operations so that data packets from the MAC are transmitted on the proper RF channel.

Channel routing: it controls the routing of data packets from the LLC to the designated channel within channel coordination operations.

User priority: this standard supports a variety of safety and non-safety applications with up to 8 levels of priority as defined by both IEEE 1609.3 and IEEE 802.11p. The user priority is used to contend for medium access using the enhanced distributed channel access functionality.

MAC service data unit (MSDU) data transfer: it controls the channel data transfer, so the WAVE short messages (WSM) conforming to the WSM protocol (WSMP) can be exchanged directly among devices on the control channel.

C. IEEE 1609.3

IEEE 1609.3 [5] and its Corrigendum 1 [6] define WAVE networking services that represents layers 3 and 4 of the OSI model. The purpose of these standards is to provide addressing and routing services within a WAVE system, enabling multiple stacks of upper layers above it and multiple lower layers below it. Upper layer support includes in-vehicle applications providing safety and convenience to their users. WAVE networking services provides also management and data services within WAVE devices.

The data plane service carries traffic generated by, or destined for, applications. It also carries traffic between management plane entities on different machines, or between management plane entities and applications. The management plane service performs system configuration and maintenance functions. Management functions employ the data plane services to pass management traffic between devices.

The Management plane consists of the following services: application registration, WBSS management, channel usage monitoring, IPv6 configuration, received channel power indicator monitoring, and MIB maintenance; as they are briefly described next:

Application registration: an application hosted in the WAVE device should be registered with the WAVE management entity, before it will be supported by management plane services, the application sends a registration request and the WAVE management entity verifies in the management information base that a unique record is being requested, then the request shall be accepted.

WAVE basic service set (WBSS) management: this service consists of the processes involved in establishing the WBSS initiated by the WAVE management entity, the management services provided are: link establishment, dynamic WBSS, WBSS credentials, WBSS completion, and application WBSS status maintenance.

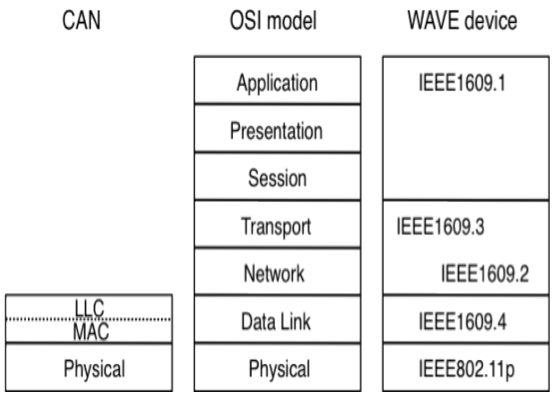

Figure 1: CAN and IEEE 1609 layers with regard to the OSI reference model

Channel usage monitoring: the WAVE management entity monitors the service channels that are used by WAVE devices nearby, so that, when called upon to do so, it can choose a WBSS service channel that is less likely to be congested. When the application requests the best available channel, the WAVE management entity would then use the least recently used a service channel.
IPv6 configuration: a network administrator provides the IPv6 information used by an RSU. The OBU derives the information that allows it to operate on an infrastructure network from a RSU connected to that network. Device link-local addresses are derived locally by any WAVE device and may be used without any external configuration information.
Received channel power indicator monitoring: a WAVE device allows an application to initiate a query of received signal strength, indicative of channel quality, at a remote device, with the measurement report returned to the requesting application.
Management information base maintenance: the WAVE management entity maintains a MIB containing the configuration and status information.

D. IEEE 1609.2

IEEE 1609.2 [7,9] describes the administrative functions necessary to support the core security functions as the authentication of control information and confidentiality services. It defines secure message formats, and their processing, within the dedicated short-range communication (DSRC)/WAVE system. The standard covers methods for securing WAVE management messages and application messages, with the exception of vehicle-originating safety messages.
The authentication service ensures that the information is received from a reliable source; authentication protects legitimate nodes from attacks like masquerading, transaction tampering, broadcast tampering, GPS spoofing, replay attack or black hole [8]. To achieve the authentication in WAVE architecture the IEEE 1609.2 proposes the usage of digital certificates. A signed message includes a certificate chain that is a list of previous digital certificates that authorized the vehicle's certificate; the first digital certificate in the certificate chain is know as the root certificate which is issued by a higher authority such as a governmental agency; through chain certificate construction the information source are validate. Every digital certificate in the certificate chain is compared against the Certificate Revocation List (CRL) to guarantee that none digital certificate has been revoked. The CRLs are constantly updated, when this occurs the node will check its storage certificates.
The confidentiality service protects data from unauthorized disclosure. The information is protected from some critical issues as inside and upside eavesdropping, and location tracking. IEEE 1609.2 [9] standard supports the Advanced Encryption Standard with Counter with CBC-MAC mode (AES – CCM mode) block cipher algorithm, with some variants: to encrypt a message with AES-CCM, the sender shall use the mechanism defined by NIST SP 800-38C. If more than one message is encrypted with the same AES-CCM key, the sender shall use a different nonce for each message. The formatting mechanism used shall be the one described in Appendix A.2 of NIST SP 800-38C; and the symmetric key used in AES-CCM shall be encrypted using the Elliptic Curve Integrated Encryption Scheme (ECIES) as specified by IEEE Std. 1363a-2004.

E. IEEE 1609.1

IEEE 1609.1 [10] is a withdrawn standard; nevertheless it describes an application that consists of the WAVE resource management (RM) and its partner the resource command processor (RCP), providing services to a remote entity, a resource management application (RMA). The RM relays orders from the RMA to the RCP. In turn, the RCP executes the commands and returns responses to the RMA through the RM. Whereas the RCP is in OBUs, the RM may reside in an OBU and RSU.
Implemented as a WAVE application, RM provides services to the RMA allowing access to both the memory and the user interfaces within the OBUs, and also connections to other on board equipment controlled by the RCP. RM acts as if it were an application layer for an RMA.
In general, RMA communicate with one or more RMs through a wired network, while an RM is communicating with an RCP on the wireless link.

IV. Gateway IEEE 1609 - CAN

The unified software development process [11] is guided by the use cases, based on the architecture, and it is iterative and incremental. This methodology allows constructing the following models: use cases, analysis, design, deployment, implementation, and test. Each model, in turn, is composed of several diagrams defined by the unified modeling language (UML).
This section presents the gateway's architecture, analysis and design models.

A. Gateway's architecture

IEEE 1609.1 standard specifies the communication between RMA and RM as well as the communication between an RM, that can be located in an RSU or an OBU, and RCP hosted in an OBU, but the communication between the RCP and the internal networks of the vehicle is outside the scope of that standard.

Therefore, this paper proposes a gateway, at the application layer of one CAN node, to serve as interface between the RCP and CAN network, as it is shown in Fig. 2.
The gateway has a table that defines all the identifiers of the CAN messages, named CAN ID table. As it was described in section II, an identifier describes the meaning of the data, so all CAN nodes decide by message filtering whether the data is processed by them or not.
In this way, the gateway will have the values of all the identifiers of the CAN messages, although the CAN ID table may be configured with only the interested identifiers, such as that of the ABS.
The gateway is composed of a CAN-IEEE 1609 protocol entity that manages the writing and reading requests from CAN network and WAVE device. This entity will be the only one who can access the CAN ID table in the gateway; through a new set of proposed primitives like read and write request and responses the gateway communicates with the RCP, in this way the interconnection between CAN and IEEE 1609 may be possible.
On the other hand in the WAVE network, an RMA, in its simplest form, retrieves data in blocks of memory that reside on the OBU. The blocks of memory are known as pages and are part of the OBU RCP resources. Some pages are accessible to all RMAs; others are registered to a specific RMA and accessible only by that RMA.
There are three types of pages as follows: the storage pages of read/write general purpose, the memory-mapped pages used as user interfaces, and the transfer pages provided as a means of interfacing with onboard equipment and networks, such as CAN.
Therefore, one transfer page shall be associated with CAN and controlled by the RCP. The authorized RMA reads and writes data from and to this page, and the data will be transferred to the associated interface.
Fig. 2 shows the communication between an application associated with an RMA and the CAN network using the RM, the RCP and the gateway.

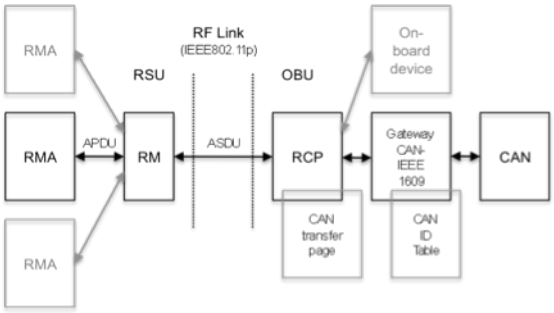

Figure 2: Interconnection between CAN and IEEE 1609

It can be noted that IEEE 1609.1 defined the communication between the RM and RCP by means of commands and responses (named ASDU in Fig. 2), such as: Reserve/Release partition, Reserve/Release memory page, Read/Write memory pages, Sleep transaction, Insert message, Set user interface. The syntax of these commands can be consulted in [10].
In the same way, IEEE 1609.1 defined the communication between RMAs and RM by means of services (named APDU in Fig. 2) such as: RMA-Activate (Request, Response), RMA-Notify (Indication, confirmation), RMA-TerminateSession (Indication, Confirmation), RMA-Exchange (Request, Response, Conformation), RMA-Deactivate (Request, Response).

B. Analysis model

This model is composed of the use cases diagram and a first approach to the class diagram. The former describes the services provided by the system to the actors (users playing a role in the system). The Fig. 3 shows the read and write services from and to CAN, which are described below.
Write to CAN: The gateway receives a write request from the RCP, the request contains the identifier of the CAN message and its value. The CAN-IEEE 1609 protocol entity serves the request and builds a CAN data frame requesting its transmission to its LLC sublayer, the entity waits for an acknowledge and after its reception writes the new value into the CAN ID table. Finally the entity sends to RCP a write confirmation.

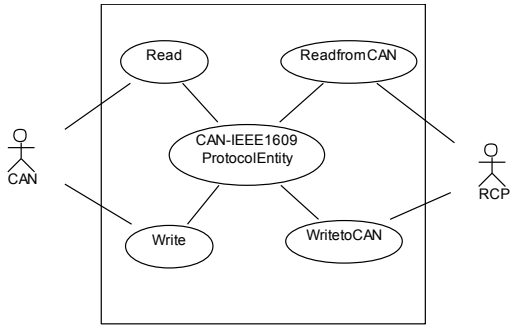

Figure 3: Use case diagram of the gateway CAN-IEEE 1609

Read from CAN: The gateway receives a read request from the RCP, the request contains the identifier of the CAN message. The CAN-IEEE 1609 protocol entity reads the value from the CAN ID table and sends the value as a read confirmation. It can be noted that CAN ID table stores the most recent values of all the identifiers.
Write: The gateway's LLC sublayer is configured to filter all the CAN data frames, i.e., for receiving all the CAN identifiers. In this way, when a CAN node transmits a data frame the CAN-IEEE 1609 protocol entity receives the new value and writes it into the CAN ID table.
Read: The gateway's LLC sublayer filters a remote frame from the CAN bus, it waits the transmission of a data frame with the same identifier by a CAN node; after reception notifies to the CAN-IEEE1609 protocol entity as a write request for update the new value into the CAN ID table.

C. Design model

This model is composed of the following UML diagrams: refined classes, sequence, collaboration, state, and activity. In order to keep compact this document this section presents only the class and sequence diagrams, the rest can be consulted in [12].
Fig. 4 shows the class diagram, it is a refined version of the class diagram of the analysis model and shows the relationships between the classes as well as their attributes and methods.

It can be noted that the gateway is composed of both the CAN-IEEE 1609 protocol entity implementing the read and write services, and the CAN ID table. This class has two relations, one with its LLC sublayer and the other with the RCP.
The class RCP defines as methods the commands required for communicating with the RM and also the methods for requesting the gateway's services.
The sequence diagrams show the objects in the system and how they interact. Fig. 5 shows the sequence diagram of the Write use case. Fig. 6 and 7 show the sequence diagrams of the WritetoCAN and ReadfromCAN use cases.
It can be noted that these services may be called when the RCP receives from RM the following commands write memory page and read memory page, which are not shown. However, a real-time analysis will be required in order to bound the latency time.

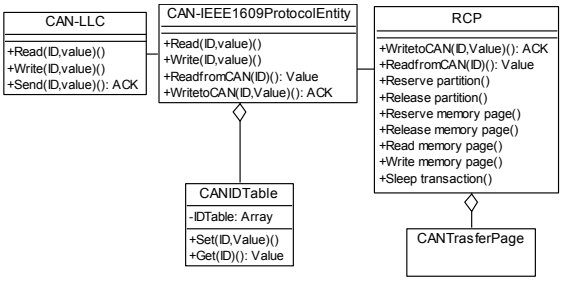

Figure 4: Class diagram of the gateway CAN-IEEE 1609

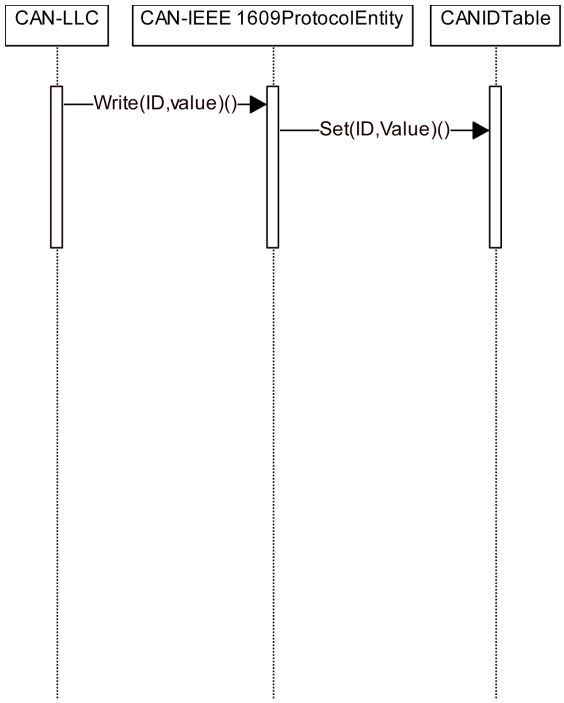

Figure 5: Sequence diagram of the Write service

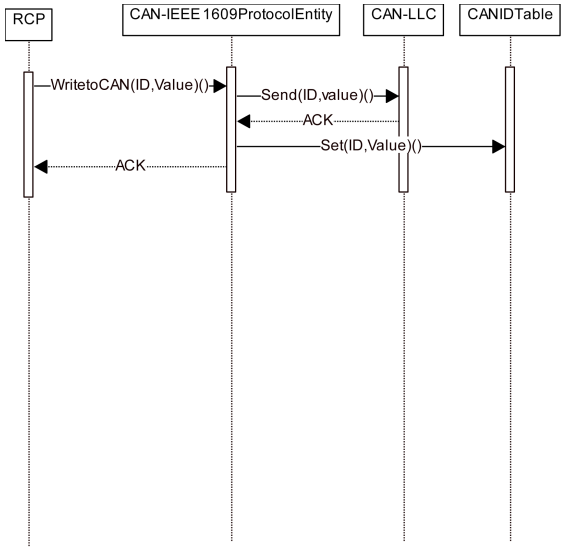

Figure 6: Sequence diagram of the WritetoCAN service

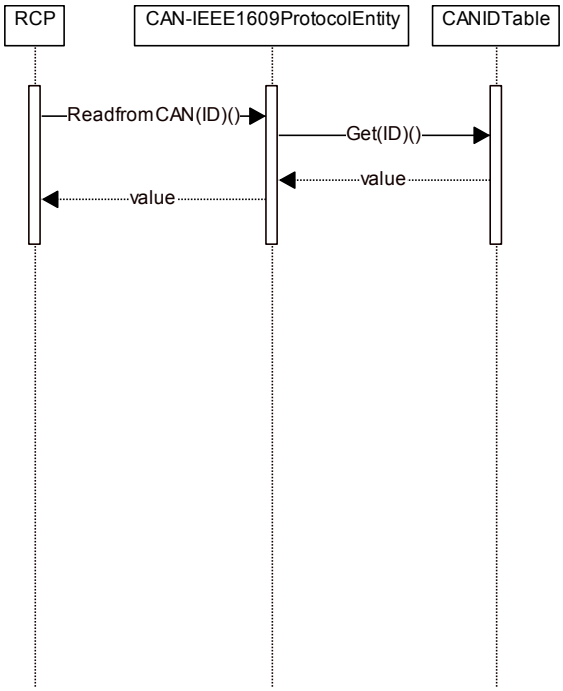

Figure 7: Sequence diagram of the ReadfromCAN service

V. Conclusion

This paper has presented the model of a gateway for the interconnection of the Controller Area Network and IEEE 1609 standards that defines Wireless Access in Vehicular Environment (WAVE) including the communications vehicle-to-vehicle and vehicle-to-infrastructure.

The goal of the gateway is to use the potential collision messages not only as warning messages for the vehicle driver but as control messages for the device controlling the brake into a vehicle. Nevertheless, the variables and their values, as well as their processing to generate the potential collisions message are our future research work.

The gateway is developed using the unified software development process and its language the unified modeling language. The paper has presented the analysis and design model. The implementation and test models are our future research work as well as the real-time and security analysis of the interconnection.

Silviana Juárez Chalini
Universidad de la Cañada
Carr. Teotitlán-San Antonio Nanahuatipan km 1.7
MX-68540 Teotitlán de Flores Magón, Oaxaca
Tel.: +52-236-3720712
E-mail: sjchalini@unca.edu.mx

Miguel Angel León Chávez
Benemérita Universidad Autónoma de Puebla
14 sur y av. San Claudio, CU, Edif. 104A.
MX-72570 Puebla
Tel.: +52-222-2295500 ext. 7213
Fax: +52-222-2295672
E-mail: mleon@cs.buap.mx

Gladys Diaz
L2TI – Institut Galilée
Université Paris 13, Sorbonne Paris Cité
Avenue J-B Clément
FR-93430 Villetaneuse
Tel.: +33-1-49404062
Fax: +33-1-49404061
gladys.diaz@univ-paris13.fr
www-l2ti.univ-paris13.fr/~diaz

References

[1] R. Bosch, GmbH. CAN Specification Version 2.0. Postfach 50, D7000 Stuttgart 1, Germany. 1991.

[2] ISO 11898-2. International Organization for Standardization. Road Vehicles – Controller Area Network (CAN) – Part 2: High Speed Medium Access Unit, 2003.

[3] IEEE Std. 802.11p. Standard for Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. Amendment 6: Wireless Access in Vehicular Environments. IEEE Computer Society, 2010.

[4] IEEE Std. 1609.4. IEEE Standard for Wireless Access in Vehicular Environments (WAVE) – Multi-channel Operation, 2010.
[5] IEEE Std. 1609.3. IEEE Standard for Wireless Access in Vehicular Environments (WAVE) – Networking Services, 2010.
[6] IEEE Std. 1609.3/Cor 1 – IEEE Standard for Wireless Access in Vehicular Environments (WAVE) – Networking Services Corrigendum 1: Miscellaneous Corrections, 2012.
[7] IEEE Std. 1609.2. IEEE Standard for Wireless Access in Vehicular Environments — Security Services for Applications and Management Messages, 2013.
[8] [C. Laurendeau, and M. Barbeau. Threats to Security in DSRC/WAVE. In T. Kunz and S.S. Ravi (Eds.): ADHOC-NOW 2006, LNCS 4104, pp. 266–279, 2006. Springer-Verlag Berlin Heidelberg, 2006.
[9] IEEE Std. 1609.2 (Withdrawn std.). IEEE Trial-Use Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages, 2006.
[10] IEEE Std. 1609.1 (Withdrawn std.). IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Resource Manager, 2006.
[11] I. Jacobson, G. Booch, and J. Rumbaugh. The Unified Software Development Process. Addison-Wesley, 1 ed. 1999.
[12] S. Juárez-Chalini. Interconnection model between CAN and IEEE 1609. Master of Science thesis. Facultad de Ciencias de la Computación – BUAP, diciembre 2009.