

**Chenxing Chi**

**DESARROLLO DE UNA PLATAFORMA ITS PARA SERVICIOS C-ITS QUE  
USE COMUNICACIONES V2X**

**TRABAJO DE FIN DE GRADO**

**Dirigido por Marc Sánchez Artigas**

**Grado de Ingeniería Informática**



**UNIVERSITAT ROVIRA I VIRGILI**

**Tarragona**

**2025**



## **Resum**

Aquest treball de final de grau té com a objectiu el desenvolupament d'una plataforma per a sistemes de transport intel·ligents (ITS) amb suport per a comunicacions cooperatives (C-ITS), basades en comunicacions entre vehicles (V2X). El projecte s'estructura en dues parts diferenciades: una fase d'estudi i una fase d'implementació.

En la primera fase es realitza un estudi tècnic sobre les plataformes ITS (Intelligent Transport Systems), amb l'objectiu de comprendre què són els sistemes C-ITS, quines tecnologies existeixen actualment per a la comunicació entre vehicles i infraestructures i quina és la situació actual del sector a nivell europeu. Aquesta fase inclou l'anàlisi dels estàndards ETSI, així com una revisió de les tecnologies de transmissió més utilitzades.

La segona fase del projecte consisteix en el disseny i la implementació d'una plataforma ITS de Applus+ IDIADA, una companyia tecnològica en el sector. En aquesta etapa, es definiran amb detall els requisits funcionals i no funcionals, així com les principals característiques de la solució. La plataforma es compon de diversos microserveis (majoritàriament desenvolupats en Java) que s'orquestran de la manera via API REST i missatgeria MQTT.

**Resumen.**

Este trabajo de fin de grado tiene como objetivo el desarrollo de una plataforma para sistemas de transporte inteligentes (ITS) con soporte para comunicaciones cooperativas (C-ITS), basadas en comunicaciones entre vehículos (V2X). El proyecto se estructura en dos partes diferenciadas: una fase de estudio y una fase de implementación.

En la primera fase se lleva a cabo un estudio técnico sobre las plataformas ITS (Intelligent Transport Systems), con el objetivo de comprender qué son los sistemas C-ITS, qué tecnologías existen actualmente para la comunicación entre vehículos e infraestructuras y cuál es la situación actual del sector a nivel europeo. Esta fase incluye el análisis de los estándares ETSI, así como una revisión de las tecnologías de transmisión más utilizadas.

La segunda fase del proyecto consiste en el diseño y la implementación de una plataforma ITS de Applus+ IDIADA, una compañía tecnológica en el sector. En esta etapa se definirán con detalle los requisitos funcionales y no funcionales, así como las principales características de la solución. La plataforma se compone de diversos microservicios (mayoritariamente desarrollados en Java) que se orquestan mediante API REST y mensajería MQTT.

**Abstract**

This final degree project aims to develop a platform for Intelligent Transport Systems (ITS) with support for Cooperative ITS (C-ITS) communications, based on Vehicle-to-Everything (V2X) messaging. The project is structured in two distinct parts: a study phase and an implementation phase.

In the first phase, a technical study of ITS platforms (Intelligent Transport Systems) is carried out in order to understand what C-ITS systems are, which technologies currently exist for vehicle-to-infrastructure and vehicle-to-vehicle communication, and what the current state of the sector is at the European level. This phase includes an analysis of ETSI standards as well as a review of the most widely used transmission technologies.

The second phase of the project consists of the design and implementation of an ITS platform for Applus+ IDIADA, a technology company in the sector. In this stage, both the functional and non-functional requirements will be defined in detail, along with the main characteristics of the solution. The platform comprises several microservices (mostly developed in Java) that are orchestrated via RESTful APIs and MQTT messaging.

# Índice

<b>1</b>	<b>INTRODUCCIÓN .....</b>	<b>6</b>
<b>2</b>	<b>INTELLIGENT TRANSPORT SYSTEM (ITS) .....</b>	<b>7</b>
2.1	INTRODUCCIÓN A ITS .....	7
2.2	DIFERENTES USOS DE ITS.....	7
<b>3</b>	<b>COOPERATIVE INTELLIGENT TRANSPORTATION SYSTEMS (C-ITS).....</b>	<b>9</b>
3.1	INTRODUCCIÓN A C-ITS .....	9
3.2	BENEFICIOS PARA LA SEGURIDAD VIAL, LA EFICIENCIA DEL TRÁFICO Y MÁS ALLÁ	9
3.3	FASES DE DESPLIEGUE DE C-ITS .....	10
3.4	COMPONENTES DE C-ITS.....	13
3.5	ENTIDADES RESPONSABLES DE LOS ESTÁNDARES C-ITS.....	14
<b>4</b>	<b>ARQUITECTURA DE UNA PLATAFORMA ITS .....</b>	<b>15</b>
4.1	INTRODUCCIÓN.....	15
4.2	CAPA DE TECNOLOGÍAS DE ACCESO (ITS ACCESS TECHNOLOGIES LAYER).....	16
4.2.1	<i>Tecnologías de comunicación, DSRC</i> .....	16
4.2.2	<i>Tecnologías de comunicación, C-V2X</i> .....	17
4.2.3	<i>Compatibilidad y requisitos regulatorios en Europa</i> .....	18
4.2.4	<i>DSRC vs C-V2X</i> .....	19
4.2.5	<i>Evolución hacia el futuro</i> .....	20
4.3	CAPA DE RED Y TRANSPORTE (ITS NETWORKING & TRANSPORT LAYER) .....	21
4.3.1	<i>GeoNetworking</i> .....	22
4.4	CAPA DE SERVICIOS (ITS FACILITIES LAYER).....	23
4.4.1	<i>ETSI common data dictionary (CDD)</i> .....	23
4.4.2	<i>Tipos de mensajes</i> .....	26
4.5	CAPA DE APLICACIÓN (ITS APPLICATIONS LAYER).....	29
<b>5</b>	<b>IMPLEMENTACIÓN DE LA PLATAFORMA ITS DE IDIADA .....</b>	<b>31</b>
5.1	INTRODUCCIÓN A LA PLATAFORMA ITS DE IDIADA .....	31
5.1.1	<i>Objetivos</i> .....	32
5.1.2	<i>Listado de tareas realizadas</i> .....	32
5.1.3	<i>Diagrama de Gantt</i> .....	32
5.2	ÁRBOL DE DEPENDENCIAS.....	35
5.3	ETSI COMMON DATA DICTIONARY (CDD) .....	36
5.3.1	<i>Data Elements</i> .....	37
5.3.2	<i>Data Frames</i> .....	39
5.4	MENSAJES ITS.....	43
5.4.1	<i>Implementación del CAM</i> .....	43
5.4.2	<i>Implementación de codificadores y descodificadores</i> .....	45
5.4.3	<i>Adaptation-Lib</i> .....	46
5.5	ITS-PLATFORM.....	47
5.5.1	<i>GeoMessaging</i> .....	48
5.5.2	<i>Geomessaging internal broker (MQTT)</i> .....	49
5.5.3	<i>Identity Manager</i> .....	53
5.5.4	<i>Message Manager</i> .....	54
5.5.5	<i>C-ITS Client</i> .....	55
5.5.6	<i>Message Extractor</i> .....	56
5.5.7	<i>External Broker</i> .....	56
5.5.8	<i>RSU Manager</i> .....	57
5.5.9	<i>External services</i> .....	58
<b>6</b>	<b>TAREAS REALIZADAS DURANTE LAS PRÁCTICAS .....</b>	<b>59</b>
6.1	CLEAN AND UPDATE MAVEN LIBRARIES IN SEVERAL PROJECTS .....	59
6.1.1	<i>Conclusión de la tarea</i> .....	59

6.2	IMPLEMENT AOP IN MESSAGE EXTRACTOR.....	60
6.2.1	<i>Introducción a AOP</i> .....	60
6.2.2	<i>Conceptos principales de AOP</i> .....	60
6.2.3	<i>Introducción a la tarea</i> .....	61
6.2.4	<i>Conclusión de la tarea</i> .....	61
6.3	INVESTIGATE CAM COMPATIBILITY BETWEEN CAMV2 AND CAMREL21 .....	62
6.4	CDD-V4 IMPLEMENTATION .....	63
6.5	GET ASN1TYPE FROM DF IN ADAPTATIONLIB NEW CAM VERSION .....	64
<b>7</b>	<b>CONCLUSIÓN</b> .....	<b>65</b>

## Índice de figuras

FIGURA 1. AWARENESS DRIVING.....	10
FIGURA 2. SENSING DRIVING.....	11
FIGURA 3. COOPERATIVE DRIVING .....	12
FIGURA 4. LOGOTIPOS DE LAS PRINCIPALES ENTIDADES .....	14
FIGURA 5. DIAGRAMA DE REFERENCIA DE UNA ESTACIÓN ITS (ETSI EN 302 665) .....	15
FIGURA 6. ARQUITECTURA DSRC.....	16
FIGURA 7. ARQUITECTURA C-V2X.....	17
FIGURA 8. IMAGEN REPRESENTATIVA DE LA COMPARATIVA DE DSRC Y C-V2X.....	19
FIGURA 9. DIAGRAMA DE REFERENCIA DE LA CAPA DE RED Y TRANSPORTE (ETSI EN 302 665).....	21
FIGURA 10. REPRESENTACIÓN DE UNICAST .....	22
FIGURA 11. REPRESENTACIÓN DE GEOBROADCAST .....	22
FIGURA 12. REPRESENTACIÓN DE TOPOLOGICALLY-SCOPED BROADCAST.....	22
FIGURA 13. TIPOS DE MENSAJES DEFINIDOS POR LAS ENTIDADES ESTANDARIZADORAS (ETSI, SAE Y ISO).....	28
FIGURA 14. EJEMPLO DE AVISO.....	29
FIGURA 15. EJEMPLO DE INTERFAZ GLOSA.....	29
FIGURA 16. EJEMPLO DE INTERFAZ IN-VEHICLE ALERTS.....	30
FIGURA 17. EJEMPLO DE AVISO (EVW) .....	30
FIGURA 19. DIAGRAMA SIMPLIFICADO DE LOS ATRIBUTOS QUE LLEVA EL CAM.....	43
FIGURA 26. DIAGRAMA REPRESENTATIVO DEL PATRÓN PUBLISH/SUBSCRIBE.....	50
FIGURA 27. REPRESENTACIÓN DEL FUNCIONAMIENTO DE QUADTILES.....	52

### **Declaración de confidencialidad**

Este trabajo de fin de grado esta realizado bajo un acuerdo de confidencialidad. El contenido incluido en este proyecto forma parte de Applus+ IDIADA. Se exige a los lectores mantener la más estricta confidencialidad respecto a esta información y no deben divulgarla ni utilizarla salvo en los casos permitidos por el acuerdo. El uso o divulgación no autorizados de esta información podrá conllevar las consecuencias estipuladas en dicho acuerdo. La información confidencial está señalada con los siguientes indicadores de inicio y fin.

**[INFORMACIÓN CONFIDENCIAL]**

[...]

**[FIN INFORMACIÓN CONFIDENCIAL]**

## **Objetivos**

El objetivo principal de este trabajo de fin de grado es comprender y analizar en profundidad los Sistemas de Transporte Inteligente (ITS) y, más específicamente, los Sistemas Cooperativos ITS (C-ITS), mediante la participación en el desarrollo, diseño e integración de una plataforma real desarrollada por Applus+ IDIADA. Con ello, se pretende aplicar los conocimientos adquiridos a lo largo del Grado en Ingeniería Informática en un entorno profesional y tecnológico actual.

Entre los objetivos concretos del trabajo se encuentra estudiar la arquitectura de las estaciones ITS definida por los estándares europeos (ETSI e ISO), así como entender el papel que desempeñan las tecnologías de comunicación V2X en escenarios cooperativos. También se buscará participar en tareas de desarrollo e integración de mensajes cooperativos (como CAM, DENM, IVIM, MAPEM, SPATEM, entre otros), hacer uso de protocolos de comunicación como MQTT y GeoNetworking, y diseñar interfaces HMI orientadas a usuarios de la vía y operadores de tráfico.

Además, el trabajo permitirá desarrollar la capacidad de adaptación a nuevas herramientas, tecnologías y entornos de desarrollo. Todo ello contribuirá a consolidar habilidades técnicas en programación, sistemas distribuidos y tratamiento de datos, así como potenciar la autonomía y la capacidad de aprendizaje continuo en contextos reales de ingeniería.

## 1 Introducción

La movilidad está experimentando una profunda transformación impulsada por el avance tecnológico, la digitalización y la creciente necesidad de sostenibilidad y seguridad. En este contexto, los Sistemas de Transporte Inteligente (ITS) se consolidan como una pieza clave para modernizar las infraestructuras de transporte y hacer frente a los retos actuales de la movilidad urbana e interurbana. Estos sistemas integran tecnologías de la información y la comunicación en el transporte terrestre con el objetivo de mejorar la eficiencia, reducir el impacto ambiental, optimizar el tráfico y, sobre todo, aumentar la seguridad vial.

La Unión Europea, consciente de la urgencia de reducir la siniestralidad en las carreteras, ha establecido el ambicioso objetivo de alcanzar la Visión Cero en 2050, con un hito intermedio claro: reducir en un 50% las muertes y lesiones graves en accidentes de tráfico para el año 2030. Esta estrategia, recogida en el Marco de Política de Seguridad Vial de la UE 2021-2030, reconoce el papel central de los ITS como herramientas fundamentales para alcanzar estos objetivos, especialmente a través de tecnologías de comunicación vehículo-a-vehículo (V2V) y vehículo-a-infraestructura (V2I), gestión inteligente del tráfico, y sistemas avanzados de asistencia a la conducción.

Este Trabajo de Final de Grado se enmarca en la estrategia europea Visión Cero 2030, cuyo objetivo es reducir drásticamente la siniestralidad vial mediante el uso de tecnologías avanzadas y una movilidad más segura, conectada y eficiente. En este contexto, los Intelligent Transport Systems (ITS) se presentan como una herramienta clave para alcanzar dicho objetivo, gracias a su capacidad para optimizar el tráfico, reducir emisiones y, sobre todo, mejorar la seguridad de todos los usuarios de la vía.

Con el fin de abordar esta temática, el trabajo se divide en dos grandes bloques. En primer lugar, se realiza un análisis exhaustivo de las tecnologías disponibles que contribuyen a mejorar el entorno del tráfico. Se estudian distintos Sistemas de Transporte Inteligente, con especial atención a aquellos que integran tecnologías de comunicación vehículo a vehículo (V2V) y vehículo a todo (V2X). Esta primera parte no solo tiene como objetivo comprender el estado actual del desarrollo tecnológico en el ámbito ITS, sino también sentar las bases para diseñar una plataforma innovadora que facilite la comunicación inteligente entre vehículos.

La segunda parte del trabajo se centra en el estudio e implementación práctica de un sistema ITS real: la plataforma cooperativa desarrollada por Applus+ IDIADA. Esta solución utiliza tecnologías CV2X y 5G para permitir que los vehículos se comuniquen entre sí y con su entorno en tiempo real. Esta experiencia práctica, desarrollada en el marco de una estancia en el equipo de Vehículos Conectados de Applus+ IDIADA, permite aplicar los conocimientos adquiridos durante la titulación en un entorno profesional, y aporta una visión concreta del potencial de los ITS en el contexto de los objetivos marcados por la Visión Cero.

## 2 Intelligent Transport System (ITS)

### 2.1 Introducción a ITS

Los Sistemas de Transporte Inteligente (ITS, por sus siglas en inglés) constituyen un conjunto integrado de tecnologías y soluciones orientadas a mejorar la eficiencia operativa, la sostenibilidad y la seguridad del transporte terrestre, con especial énfasis en la red vial, aunque su aplicación se extiende también al ámbito ferroviario. Estas soluciones se enmarcan en el paradigma del Internet de las Cosas (IoT), dado que se basan en la interconexión de dispositivos físicos capaces de recolectar, intercambiar y procesar datos en tiempo real.

Mediante la conexión entre vehículos, señales de tráfico, estaciones de peaje e infraestructuras viales, los ITS permiten optimizar la gestión del tráfico, reducir los tiempos de desplazamiento, prevenir accidentes y disminuir el impacto ambiental asociado al transporte. El crecimiento exponencial del parque automovilístico en las últimas décadas ha intensificado la congestión en las vías urbanas e interurbanas, lo que a su vez ha incrementado el consumo energético y las emisiones contaminantes, agravando los problemas medioambientales y de movilidad.

Ante este contexto, diversos países y ciudades han comenzado a implementar sistemas ITS como respuesta a la necesidad de modernizar la gestión del tráfico y avanzar hacia un modelo de movilidad más sostenible y resiliente. Además de mejorar la eficiencia del sistema de transporte, estos sistemas ofrecen un importante valor añadido en términos de seguridad vial, al proporcionar herramientas capaces de anticiparse a situaciones de riesgo mediante el análisis y la interpretación de grandes volúmenes de datos generados en tiempo real.

### 2.2 Diferentes usos de ITS

Los Sistemas de Transporte Inteligente incorporan múltiples tecnologías que permiten optimizar distintos aspectos del transporte y la movilidad. A continuación, se detallan algunas de las principales aplicaciones implementadas en la actualidad:

- **Sistemas electrónicos de cobro de peaje:** La implementación de peajes electrónicos, permite realizar el pago sin necesidad de detener el vehículo ni reducir su velocidad. Esta tecnología disminuye significativamente los tiempos de espera y elimina la congestión asociada a los métodos tradicionales de cobro. Su aplicación es frecuente en vías interurbanas, aunque también se ha extendido a entornos urbanos con tarifas dinámicas que varían según el nivel de congestión.
- **Gestión y análisis de datos en tiempo real:** Una de las funcionalidades clave de los ITS es la capacidad de recopilar, procesar y difundir información en tiempo real sobre el estado del tráfico, el transporte público y las condiciones de la infraestructura vial. Esto permite a los usuarios seleccionar rutas más rápidas y seguras, reducir los tiempos de desplazamiento y minimizar el riesgo de accidentes, contribuyendo a una circulación más fluida y eficiente.
- **Sistemas inteligentes de aparcamiento:** A través del uso de plataformas basadas en la nube, es posible conocer en tiempo real la disponibilidad de plazas de aparcamiento. Esta funcionalidad permite una gestión más eficiente del espacio urbano, reduce la circulación innecesaria en busca de aparcamiento y contribuye al ahorro de tiempo, combustible y emisiones contaminantes.

- **Seguridad vial activa:** Los sistemas avanzados de asistencia a la conducción (ADAS), como el aviso de colisión frontal, detección de ángulo muerto o frenado automático de emergencia, se integran dentro del ecosistema ITS. A nivel de infraestructura, los ITS pueden detectar condiciones peligrosas y alertar a los conductores antes de que ocurran incidentes.
- **Sistema de emergencias e-call:** Algunos vehículos integran sistemas capaces de detectar un accidente y emitir una alerta automática a los servicios de emergencia, incluyendo la localización precisa del incidente. Esta capacidad de respuesta rápida incrementa la eficiencia operativa de los equipos de rescate y mejora la seguridad de los ocupantes.
- **Vigilancia automatizada de infracciones:** Los ITS también han facilitado la implementación de sistemas automáticos para la detección de infracciones, como el exceso de velocidad o el cruce de semáforos en rojo. Estos sistemas, conectados a bases de datos centralizadas, permiten una gestión eficaz de las sanciones y han demostrado ser herramientas útiles para reforzar la seguridad vial.
- **Supervisión del impacto ambiental:** Algunos ITS están diseñados para monitorear las emisiones de gases contaminantes generadas durante un trayecto. Esta información permite a los conductores tomar conciencia sobre el impacto ambiental de sus desplazamientos y favorece la adopción de hábitos de conducción más sostenibles.

### **3 Cooperative Intelligent Transportation Systems (C-ITS)**

#### **3.1 Introducción a C-ITS**

Los Sistemas de Transporte Inteligente Cooperativos, conocidos como C-ITS (Cooperative Intelligent Transport Systems), representan una evolución dentro del ámbito de los Sistemas de Transporte Inteligente (ITS).

Mientras que los Sistemas de Transporte Inteligente (ITS) son la integración de las tecnologías de la información y la comunicación con la infraestructura del transporte, los vehículos y los usuarios, los Sistemas de Transporte Inteligente Cooperativos (C-ITS) son la parte de los ITS que se ocupa del intercambio de información en tiempo real entre los usuarios de la vía y la infraestructura, lo que permite aplicaciones y servicios avanzados con un enorme potencial para aumentar la comodidad, la movilidad y la seguridad. Esta cooperación se basa en el intercambio continuo de datos entre entidades mediante comunicaciones vehículo-a-vehículo (V2V), vehículo-a-infraestructura (V2I), infraestructura-a-vehículo (I2V) y vehículo-a-todo (V2X, vehicle-to-everything).

El objetivo principal de los C-ITS es crear un entorno de movilidad conectado, donde todos los actores puedan compartir información crítica en tiempo real para anticipar riesgos, coordinar movimientos y optimizar la toma de decisiones. Esta tecnología permite, por ejemplo, que un vehículo pueda recibir una alerta temprana sobre una frenada brusca de otro vehículo más adelante, la presencia de obras en la vía o cambios en los semáforos antes de que el conductor los perciba visualmente.

#### **3.2 Beneficios para la seguridad vial, la eficiencia del tráfico y más allá**

Los sistemas cooperativos V2X instalados en los vehículos analizan los datos recibidos y advierten al conductor de posibles peligros, por ejemplo, si se aproxima al final de una retención, una zona de obras, o si el vehículo que le precede realiza una frenada brusca. De esta manera, los sistemas V2X cooperativos favorecen una conducción preventiva, muestran riesgos que podrían no ser visibles para el conductor y contribuyen a evitar accidentes. Para el usuario, el intercambio de datos transcurre de forma completamente transparente, hasta que surge una situación de peligro. También será posible habilitar la cooperación activa entre usuarios de la vía, como en maniobras de incorporación a carriles, a través de conjuntos de mensajes estandarizados futuros.

El intercambio de datos cooperativo V2X entre los usuarios de la vía y la infraestructura funciona en cualquier lugar y momento donde se aproximen usuarios cooperativos. A medida que disminuye la distancia entre dichos usuarios, la calidad de la comunicación V2X cooperativa mejora. Bajo este principio, se pueden prevenir situaciones críticas de seguridad vial y los accidentes derivados de estas. Tanto los vehículos conducidos manualmente como aquellos equipados con sistemas de asistencia o automatización se benefician de estas tecnologías.

Servicios como la provisión de información en tiempo real sobre las fases semafóricas y sus cambios previstos, o sobre obstáculos en la ruta, también favorecen una circulación más fluida y confortable. Al evitar aceleraciones y deceleraciones bruscas, se logra reducir el consumo de combustible o energía, lo que repercute positivamente en la disminución del ruido y las emisiones contaminantes. Sensores avanzados, tanto en vehículos como en la infraestructura vial, son capaces de detectar a otros usuarios de la carretera. Los servicios V2X cooperativos también están vinculados con mejoras en la seguridad de los usuarios vulnerables de la vía (VRUs), el confort en la conducción, la eficiencia del tráfico y el impacto medioambiental del transporte.

### 3.3 Fases de despliegue de C-ITS

El Consorcio CAR 2 CAR ha desarrollado un plan de ruta de servicios y casos de uso para el despliegue gradual de C-ITS en Europa. Esta hoja de ruta está dividida en 3 fases:

**Awareness Driving (2020-2025):** El intercambio de datos de estado a través de la comunicación cooperativa V2X, como la posición, velocidad, dirección de conducción o incidentes especiales como defectos de vehículos, permite una serie de servicios de información y advertencia. Estos servicios apoyan a los usuarios de la vía al conducir de manera anticipada y hacerlos conscientes de riesgos potenciales que aún no son visibles para ellos. Ejemplos de estos servicios incluyen:

- **Advertencia de colisión en intersecciones**
- **Advertencia de vehículo de emergencia**
- **Advertencia de situación peligrosa**
- **Advertencia de vehículo estacionario**
- **Advertencia de congestión de tráfico**
- **Advertencia previa o posterior a un choque**

Esta información facilita una conducción más segura y permite que los conductores estén mejor preparados ante posibles peligros, incluso antes de que estos sean percibidos visualmente.

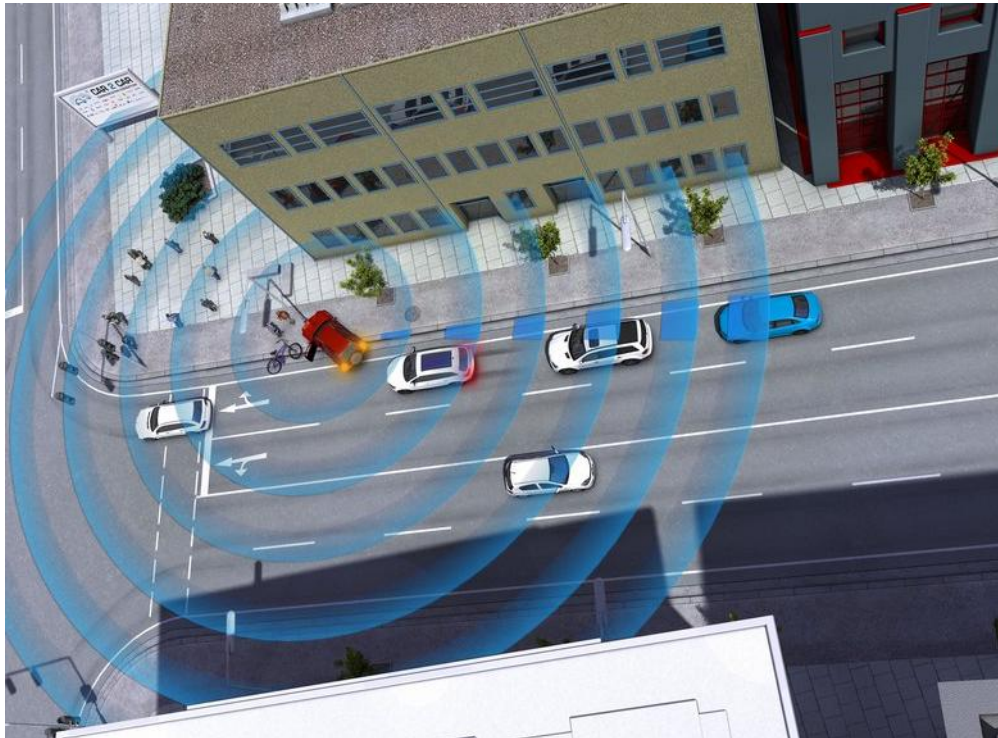
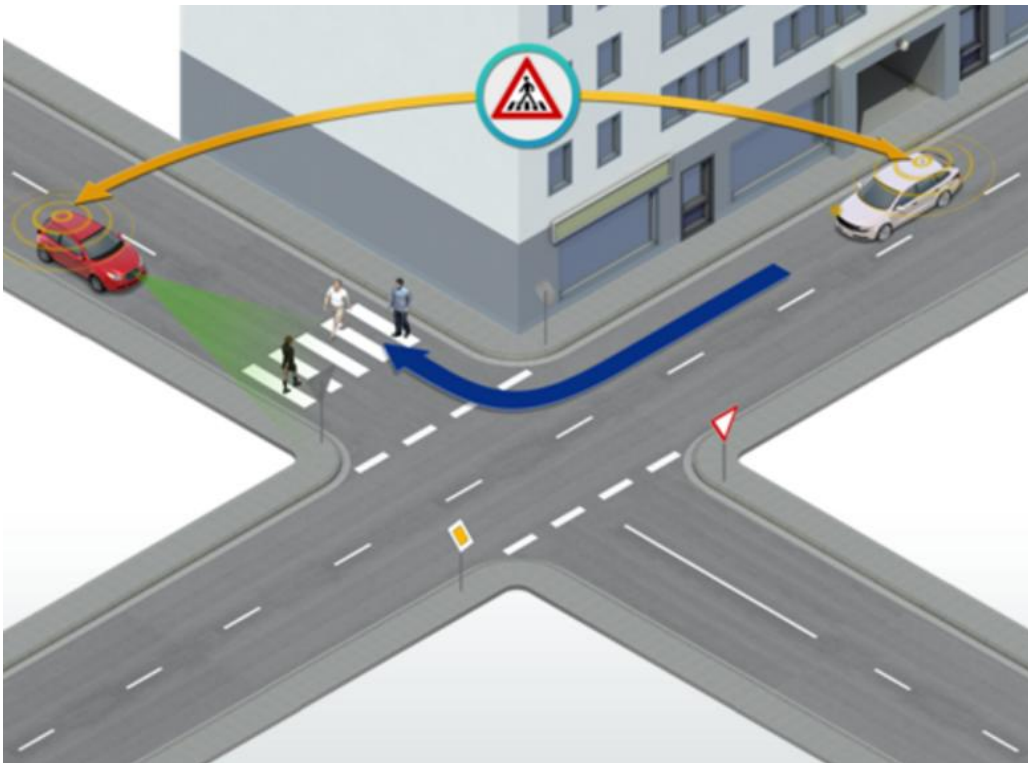


Figura 1. Awareness Driving

**Sensing Driving (2025-2030):** Además de los datos de estado, los usuarios de la vía capaces de comunicación cooperativa V2X pueden compartir observaciones obtenidas por sensores y avanzar en la recopilación de información ambiental. De este modo, no solo se advierte a los participantes del tráfico sobre peligros que aún no pueden percibir, sino que también se tiene en cuenta y se protege a los usuarios de la vía que no están equipados con sistemas de comunicación en diferentes situaciones de tráfico. Ejemplos de estos servicios incluyen:

- **Advertencia de adelantamiento**
- **Advertencia extendida de colisión en intersecciones**
- **Advertencia de usuario vulnerable de la vía**
- **Control de cruceo adaptativo cooperativo**
- **Advertencia a largo plazo de obras viales**
- **Priorización de vehículos especiales**

Estos servicios mejoran la percepción del entorno de los conductores y aumentan la seguridad vial, al integrar información de sensores y facilitar la protección de los usuarios más vulnerables.



**Figura 2.** Sensing Driving

**Cooperative Driving (2030+):** Además de los datos de estado y los datos de sensores, los usuarios de la vía capaces de comunicación cooperativa V2X también pueden proporcionar datos de intención, lo que les permite interactuar de manera inteligente y coordinar su comportamiento incluso en situaciones de tráfico complejas. La predicción del comportamiento esperado de todos los usuarios de la vía es un requisito importante para el objetivo a largo plazo de una conducción altamente automatizada y autónoma. Ejemplos de estos servicios incluyen:

- **Formación de convoyes (Platooning) estáticos o dinámicos**
- **Reserva de área**
- **Fusión cooperativa**
- **Cambio de carril cooperativo**
- **Adelantamiento cooperativo**

Estos servicios permiten una interacción fluida entre los vehículos y una mejor coordinación en el tráfico, lo que es fundamental para la evolución hacia la conducción autónoma y la mejora de la eficiencia del tráfico.



**Figura 3.** Cooperative driving

### 3.4 Componentes de C-ITS

El principal propósito de los Sistemas de Transporte Inteligente Cooperativos (C-ITS) es proporcionar a los vehículos información contextual y relevante en el momento preciso, proveniente de la infraestructura vial y de otros usuarios de la vía. A diferencia de la señalización tradicional —que en ocasiones puede resultar irrelevante para ciertos conductores—, los C-ITS filtran y transmiten únicamente los datos que afectan directamente al vehículo en cuestión. Esta información puede emplearse tanto para advertir al conductor como para activar respuestas automáticas del propio sistema, incrementando así la seguridad y la eficiencia en la conducción.

Además, los C-ITS permiten la comunicación directa entre vehículos (V2V), lo cual facilita la cooperación entre usuarios de la vía en tiempo real, incluso en situaciones complejas. Para lograr esta interacción efectiva, una plataforma C-ITS se compone de tres elementos fundamentales que permiten el intercambio y procesamiento de esta información:

**OBU:** On-Board-Unit, dispositivos instalados en los vehículos que permiten la comunicación V2X (vehículo a todo), incluyendo V2V (vehículo a vehículo), V2I (vehículo a infraestructura) y V2P (vehículo a peatón). Estas unidades recogen datos del entorno (posición, velocidad, dirección, sensores, etc.) y los transmiten a otros usuarios o sistemas.

**RSU:** Roadside-Unit, dispositivos situados en la infraestructura vial, como semáforos, señales de tráfico o paneles de información. Se encargan de emitir datos relevantes al tráfico (fases semafóricas, límites de velocidad, obras en carretera) y de recibir información de los vehículos para redistribuirla a otros usuarios.

**TMC:** Traffic Management Center, plataforma centralizada que recopila, procesa y analiza la información procedente de los distintos elementos C-ITS. Coordina decisiones sobre gestión del tráfico, incidentes y servicios adicionales, y puede distribuir instrucciones o advertencias a los usuarios.

### 3.5 Entidades responsables de los estándares C-ITS

Las principales entidades encargadas del desarrollo y la estandarización de los Sistemas de Transporte Inteligente Cooperativos (C-ITS) son las siguientes:

**CAR 2 CAR Communication Consortium (C2C-CC):** Consorcio europeo formado por fabricantes de vehículos y proveedores de tecnología que trabaja para garantizar la interoperabilidad entre vehículos de diferentes marcas. Desarrolla perfiles de comunicación y estructuras de mensajes comunes, basados en los estándares de ETSI e ISO.

**Comisión Europea:** Institución política encargada de coordinar la implementación de los C-ITS en Europa. Ha impulsado iniciativas como la C-ITS Deployment Platform y estrategias como la EU Strategy on Cooperative, Connected and Automated Mobility (CCAM), proporcionando un marco legislativo y técnico para su despliegue.

La **ETSI** (European Telecommunications Standards Institute), la **ISO** (International Organization for Standardization) y la **SAE International** (Society of Automotive Engineers) desempeñan un papel clave en la definición de los estándares que rigen el diseño y funcionamiento de estos sistemas. Estas entidades colaboran para establecer un marco normativo común que abarque:

- **La arquitectura funcional** que debe adoptar un sistema ITS.
- **Los diferentes conjuntos de mensajes** (message sets) que los sistemas deberán ser capaces de generar, enviar, recibir y diseminar, garantizando así la interoperabilidad entre actores del ecosistema V2X (vehículo a todo).
- **Los estándares de seguridad**, fundamentales para proteger las comunicaciones y evitar manipulaciones o accesos no autorizados.

Gracias a esta colaboración, se asegura una estandarización global que permite la compatibilidad entre dispositivos y servicios C-ITS desarrollados por distintos fabricantes y desplegados en diferentes regiones.



Figura 4. Logotipos de las principales entidades

## 4 Arquitectura de una plataforma ITS

### 4.1 Introducción

Para garantizar una comunicación eficiente, interoperable y segura entre los distintos actores que conforman un entorno C-ITS, es imprescindible disponer de una arquitectura de red bien estructurada. El modelo de protocolos utilizado en una estación ITS, tal y como se establece en el estándar **ETSI EN 302 665**. Se estructura en varias capas funcionales que permiten la comunicación, gestión y operación eficiente de los sistemas de transporte inteligente cooperativos. Las tres capas principales del stack de protocolos son:

- **Capa de tecnologías de acceso (Access Layer):** Se encarga de la comunicación a nivel físico y de enlace de datos. Incluye medios como el ITS-G5, que permite comunicaciones de corto alcance entre vehículos e infraestructuras. Esta capa soporta tanto comunicaciones internas entre componentes de una estación, como comunicaciones externas con otras estaciones ITS.
- **Capa de red y transporte (Networking & Transport Layer):** Abarca los protocolos responsables del envío, enrutamiento y entrega de datos entre estaciones y hacia otras redes como Internet. Aquí se integran protocolos como GeoNetworking, diseñado para comunicaciones ad hoc basadas en localización geográfica, y protocolos IP como IPv6. También se incluyen protocolos de transporte como UDP o TCP, según las necesidades de cada aplicación.
- **Capa de servicios (Facilities Layer):** Proporciona funcionalidades intermedias para dar soporte a las aplicaciones ITS. Ofrece servicios como agregación de datos, manejo de mensajes cooperativos, direccionamiento a aplicaciones y gestión de servicios. Esta capa actúa como un puente entre las aplicaciones y las capas de red.

Por encima de estas capas se sitúan las **aplicaciones ITS**. Estas aplicaciones utilizan la pila completa de protocolos para intercambiar información relevante con otras estaciones ITS o sistemas centrales. Además, el sistema incluye dos entidades transversales:

- **Entidad de gestión (Management Entity)**, que permite la configuración del sistema y la coordinación entre capas.
- **Entidad de seguridad (Security Entity)**, que garantiza la integridad, autenticación y confidencialidad de los datos transmitidos, protegiendo tanto el sistema como la privacidad de los usuarios.

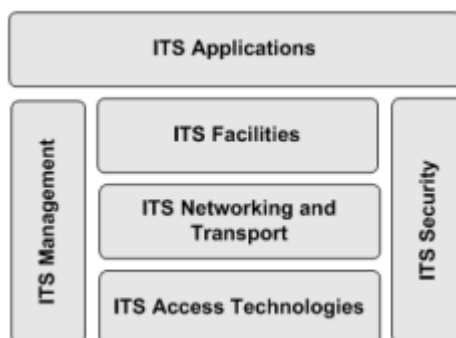


Figura 5. Diagrama de referencia de una estación ITS (ETSI EN 302 665)

## 4.2 Capa de Tecnologías de Acceso (ITS Access Technologies Layer)

Para lograr esta comunicación directa entre vehículos e infraestructuras, hoy en día existen principalmente dos tecnologías: DSRC (Dedicated Short-Range Communications), basada en el estándar IEEE 802.11p, y C-V2X (Cellular V2X), que sigue las especificaciones desarrolladas por 3GPP para redes LTE y 5G.

Estas dos tecnologías se sitúan dentro de la capa de acceso de una estación ITS, ya que permiten establecer comunicaciones locales de corto alcance de forma rápida y fiable. En este apartado vamos a profundizar en cómo funciona cada una de ellas.

### 4.2.1 Tecnologías de comunicación, DSRC

DSRC es una tecnología de comunicación inalámbrica de corto alcance dedicada, diseñada específicamente para entornos vehiculares. En Europa se la conoce como ITS-G5, estandarizada por ETSI y basada en la norma IEEE 802.11p (WLAN extendido para entornos vehiculares). Opera en la banda de 5,9 GHz reservada para ITS y permite comunicaciones directas ad hoc entre vehículos (V2V) y entre vehículos e infraestructura (V2I) sin necesidad de una red celular ni emparejamientos previos. DSRC/ITS-G5 ofrece una comunicación directa de baja latencia y alcance medio (~300 m típicamente, con alcances mayores en entornos despejados), que es adecuada para alertas de seguridad inmediatas entre vehículos.

Una característica de DSRC/ITS-G5 es que forma redes locales auto-organizadas, donde cada vehículo o estación ITS transmite mensajes de forma periódica (p. ej. diez veces por segundo) anunciando su posición, velocidad, etc., mientras escucha mensajes de los demás. DSRC utiliza un protocolo de acceso de tipo WAVE (Wireless Access in Vehicular Environments) similar al WiFi, lo que permite un funcionamiento distribuido sin coordinador central. Esto le otorga independencia de redes comerciales (no requiere suscripción ni cobertura celular) y comunicación gratuita de punto a punto. La tecnología ha sido probada durante más de una década en entornos reales, demostrando su madurez y robustez para aplicaciones de seguridad cooperativa (por ejemplo, pruebas de ETSI con cientos de vehículos).

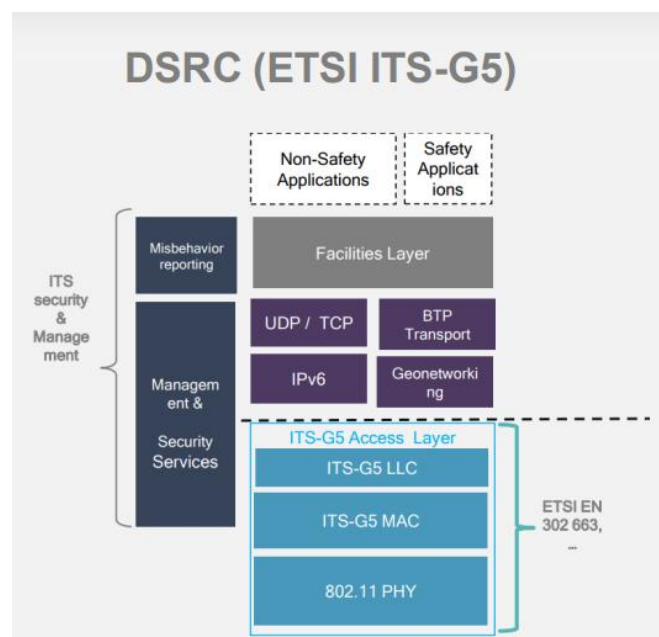


Figura 6. Arquitectura DSRC

#### 4.2.2 Tecnologías de comunicación, C-V2X

C-V2X (Cellular Vehicle-to-Everything) es la tecnología de comunicaciones V2X definida por 3GPP que aprovecha la plataforma celular (4G LTE y su evolución 5G NR) para habilitar comunicaciones vehiculares. A diferencia de DSRC, C-V2X tiene dos modos complementarios de comunicación:

**Modo directo:** comunicación *vehículo-a-vehículo* (V2V), *vehículo-a-infraestructura* (V2I) o *vehículo-a-persona* (V2P) directa mediante interfaz PC5 (también llamada sidelink). Este modo no requiere cobertura de red móvil ni una SIM, ya que los dispositivos se comunican ad-hoc en la banda de 5,9 GHz de ITS. Está diseñado para ofrecer latencia muy baja y alta fiabilidad, similar en concepto a DSRC. La versión inicial de este modo se introdujo en LTE Rel-14 (año 2017), a menudo referida como *LTE-V2X*. Técnicamente, LTE-V2X PC5 emplea transmisiones periódicas en sub-tramas LTE (1 ms) con programación semipersistente distribuida, permitiendo a cada vehículo reservar recursos de radio para sus mensajes para reducir colisiones.

**Modo de red asistido:** comunicación *vehículo-a-red* (V2N) a través de la interfaz Uu (la conexión convencional 4G/5G a estaciones base celulares). En este modo, la información vehicular se envía a servidores o a la nube a través de la red celular, y puede redistribuirse a otros vehículos. También la red celular puede asistir la coordinación de las comunicaciones directas. Este modo permite aprovechar la infraestructura móvil existente para ampliar el alcance de las aplicaciones (ej. notificar riesgos más allá del radio inmediato vía la nube) y para soportar casos de uso no estrictamente locales.

C-V2X en su primera encarnación (LTE-V2X) opera en la misma banda 5,9 GHz ITS que ITS-G5, con especificaciones de capa física definidas por 3GPP y adaptadas por ETSI para cumplir requisitos europeos. Esto permite que un solo módulo proporcione tanto conectividad a la red (4G/5G) como comunicaciones directas de corto alcance. Cabe destacar que los mensajes y protocolos de nivel superior (p. ej. mensajes de Cooperatividad CAM/DENM en Europa) son independientes de la tecnología de acceso, por lo que un mensaje V2X puede transmitirse vía ITS-G5 o vía C-V2X indistintamente a nivel de aplicaciones.

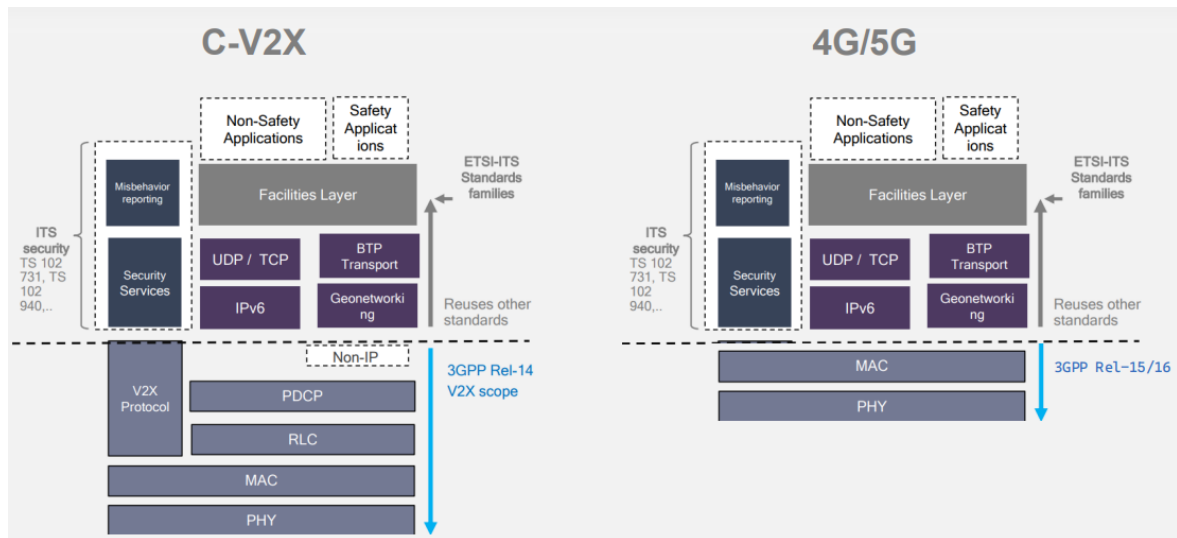


Figura 7. Arquitectura C-V2X

### **4.2.3 Compatibilidad y requisitos regulatorios en Europa**

En Europa, tanto DSRC/ITS-G5 como C-V2X son tecnologías permitidas dentro del marco regulatorio actual para C-ITS. La Comisión Europea armonizó la banda 5,875–5,905 GHz para aplicaciones ITS de seguridad vial mediante la Decisión 2008/671/EC, sin especificar una tecnología particular. En consecuencia, la regulación europea de espectro se considera “neutral en cuanto a la tecnología”, otorgando *derechos equitativos* de operación en 5,9 GHz a cualquier tecnología ITS que cumpla las condiciones técnicas requeridas. Esto significa que ITS-G5 y C-V2X pueden coexistir legalmente en las mismas frecuencias, siempre que los dispositivos respeten los límites de potencia, máscara de espectro y demás requisitos estipulados en estándares armonizados (como ETSI EN 302 571 para equipos ITS de 5 GHz).

Europa exige que ambas tecnologías sean seguras y no causen interferencias perjudiciales, pero permite su uso dentro de la banda ITS. Los requisitos incluyen certificación conforme a normas ETSI, y asegurar mecanismos de control de congestión (tanto ITS-G5 como C-V2X disponen de métodos para reducir carga de canal cuando aumenta el tráfico de mensajes). La compatibilidad a nivel de mensajes se garantiza mediante estándares comunes, de modo que un mensaje de advertencia tenga el mismo formato independientemente de si fue transmitido por DSRC o por C-V2X. No obstante, no existe interoperabilidad radio directa entre DSRC y C-V2X – un radio DSRC no puede “entender” una señal C-V2X y viceversa, por lo que el escenario mixto presenta retos: durante la transición tecnológica podrían ser necesarios equipos duales que incorporen ambas radios para comunicarse con todos los vehículos. Este aspecto sigue siendo objeto de discusión en foros estándares y regulatorios.

#### 4.2.4 DSRC vs C-V2X

- **Latencia:** Ambas tecnologías ofrecen una latencia muy baja, adecuada para aplicaciones de seguridad vial. DSRC destaca por su inmediatez gracias al acceso distribuido, mientras que C-V2X logra latencias comparables con una gestión más estructurada, y mejora aún más con 5G.
- **Capacidad de transmisión:** C-V2X tiene mayor eficiencia y escalabilidad en cuanto a volumen de datos y densidad de vehículos. DSRC funciona bien para mensajes cortos, pero puede saturarse antes en entornos muy cargados.
- **Cobertura:** C-V2X ofrece algo más de alcance y mejor comportamiento en entornos sin línea de visión (NLOS). DSRC tiene buen rendimiento en distancias cortas y medias.
- **Interoperabilidad:** Ambas son compatibles a nivel de mensajes, pero no pueden comunicarse entre sí directamente. Es necesaria infraestructura mixta o radios duales para evitar problemas en entornos híbridos.
- **Coste de despliegue:** DSRC es más barato en términos de hardware inicial. C-V2X puede aprovechar la conectividad LTE/5G ya presente en muchos vehículos, lo que a medio plazo puede equilibrar los costes.
- **Escalabilidad futura:** C-V2X tiene más recorrido evolutivo gracias a su integración con 5G y al respaldo del 3GPP. DSRC también evoluciona (con 802.11bd), pero con más limitaciones en espectro y adopción global.
- **Seguridad:** Ambas cumplen con los estándares europeos de seguridad (ETSI). C-V2X añade además la seguridad nativa de las redes móviles como complemento.



Figura 8. Imagen representativa de la comparativa de DSRC y C-V2X

#### 4.2.5 Evolución hacia el futuro

A futuro, es posible que DSRC y C-VX coexistan durante un tiempo, pero las tendencias regulatorias y de mercado podrían inclinar la balanza hacia un enfoque predominante

La mayor ventaja de C-V2X es su capacidad para aprovechar todas las funciones que ofrece la red LTE existente, lo cual permite dar soporte a todos los casos de uso actuales y futuros de los sistemas C-ITS. El 3GPP ya está trabajando en la siguiente evolución de C-V2X, que estará preparada para abordar casos de uso de seguridad más avanzados, con mejoras en el ancho de banda, la fiabilidad, la precisión en el posicionamiento y una latencia aún más baja. Gracias al ecosistema 3GPP, la evolución de C-V2X tiene un camino mucho más prometedor que el de 802.11p, lo que ampliará aún más la diferencia entre ambas tecnologías.

C-V2X ofrece una solución nativa que puede implementarse en una amplia variedad de dispositivos, incluidos los smartphones, lo cual permite que peatones, ciclistas y otros usuarios vulnerables de la vía —así como conductores de vehículos sin conectividad avanzada— puedan beneficiarse también de las ventajas de seguridad que ofrecen los ITS.

Estudios recientes han mostrado que C-V2X tiene un mejor rendimiento que 802.11p en cobertura, ofreciendo al menos un 56 % más de rendimiento, según los resultados de un estudio reciente. Además, LTE tiene una mayor eficiencia espectral, lo que le permite servir a un número más elevado de usuarios en el mismo espectro de frecuencia. También ofrece un nivel de seguridad superior en comparación con 802.11p.

Uno de los principales inconvenientes de 802.11p es que no está preparado para el futuro. No podrá adaptarse a las generaciones futuras de sistemas C-ITS basados en 5G, como los que están siendo diseñados por grupos como 5GAA, 5GCAR y otros. Esto significa que los fabricantes (OEM) se verán obligados a mantener múltiples tecnologías a la vez, lo que aumentará la complejidad de los sistemas, la integración, las pruebas y los costes en general.

A finales de 2017, se estimaba que 23 millones de coches en Europa estarían equipados con un chipset LTE (4G) para servicios de infoentretenimiento o emergencias (según datos de Machina Research). Se espera que esa cifra aumente hasta 258 millones para 2025, lo que implica que prácticamente todos los vehículos nuevos en Europa contarán con tecnología LTE integrada para entonces.

Además, desplegar 802.11p implicaría desarrollar una nueva infraestructura desde cero, lo cual complica enormemente el modelo de negocio para aplicaciones V2I (vehículo a infraestructura). Esta es una de las razones por las que esta tecnología ha estado prácticamente parada en Estados Unidos durante quince años, sin planes de implementación a gran escala. Incluso operadores de carreteras, como los de España, han mostrado su preocupación por los altos costes de operación y mantenimiento de las redes viales asociadas a esta tecnología.

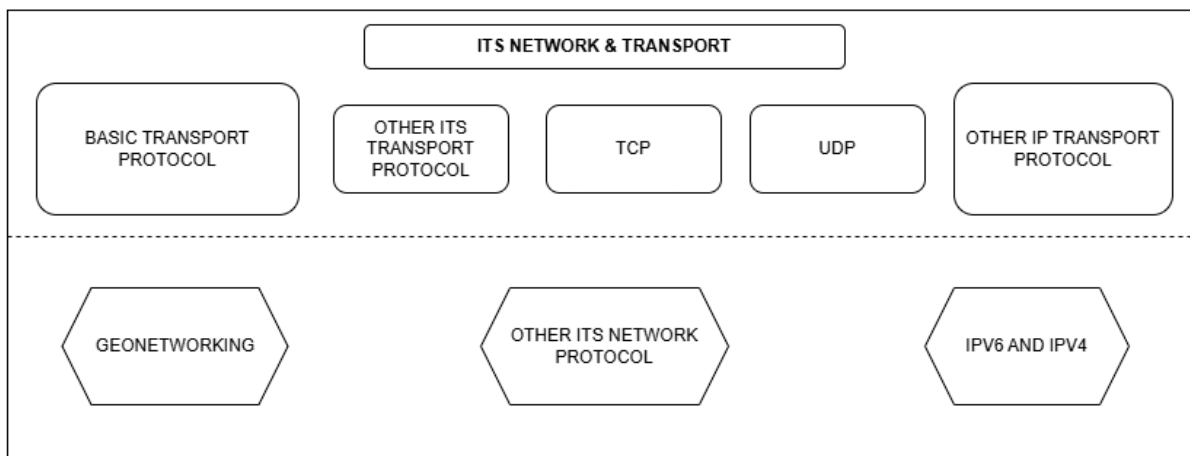
En contraste, C-V2X puede aprovechar la infraestructura LTE ya existente, lo que permitiría un despliegue mucho más rápido y a gran escala en Europa. Incluso si es necesario instalar unidades en carretera (RSUs), el hecho de contar con una red de base ya disponible reduce considerablemente los tiempos y costes de despliegue.

### 4.3 Capa de Red y Transporte (ITS Networking & Transport Layer)

La capa de red y transporte dentro de una estación ITS tiene como función principal garantizar la entrega de datos entre estaciones, así como con nodos externos a la red ITS, como pueden ser servidores centrales o redes globales como Internet. Esta capa utiliza distintos protocolos de red y transporte, que pueden operar sobre diversas tecnologías de acceso (como ITS-G5 o C-V2X). Los más destacables son los siguientes:

- **GeoNetworking:** Protocolo diseñado para permitir la comunicación geográfica entre vehículos (V2V) y entre vehículos e infraestructura (V2I). Su especificación se divide en una parte independiente del medio físico (común a cualquier tecnología) y otra parte dependiente del medio, como en el caso de ITS-G5.
- **IPv6:** Permite la transmisión de paquetes IP dentro de redes ITS.
- **IPv4:** Puede emplearse como tecnología de transición para asegurar compatibilidad con sistemas que aún no soportan IPv6.
- **Protocolos de transporte de Internet:**
  - **UDP (User Datagram Protocol).**
  - **TCP (Transmission Control Protocol).**

Si bien estos protocolos están diseñados para funcionar sobre distintas tecnologías de acceso ITS, su uso puede verse restringido en función de la tecnología concreta implementada, ya que no todas ofrecen el mismo nivel de soporte para cada protocolo.



**Figura 9.** Diagrama de referencia de la capa de red y transporte (ETSI EN 302 665)

### 4.3.1 GeoNetworking

Uno de los componentes más importantes en esta capa de red y transporte es el protocolo de GeoNetworking. Esta, expone 3 paradigmas:

- **UniCast:** envío punto a punto entre dos estaciones ITS (de un vehículo a otro en cadena).

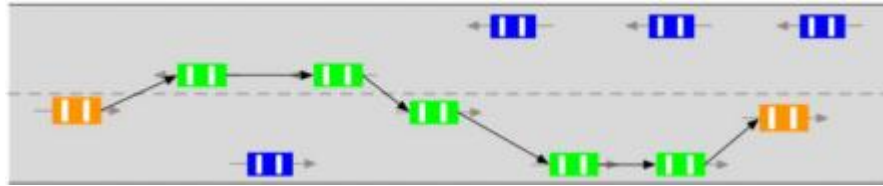


Figura 10. Representación de unicast

- **GeoBroadCast:** envío de mensajes a todas las estaciones (RSU y OBUS) dentro de una zona geográfica específica. Esta zona puede ser de área circular, rectangular, elíptica o cuadrada.

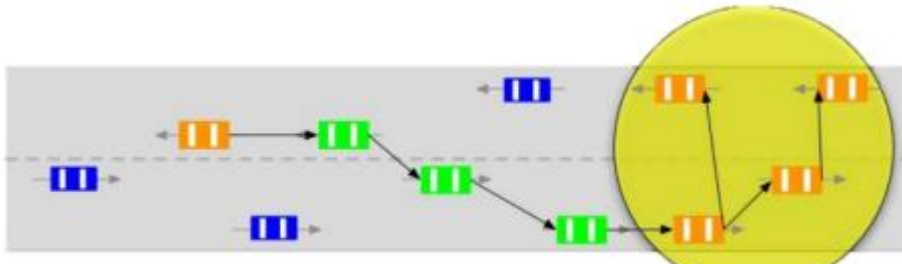


Figura 11. Representación de GeoBroadcast

- **Topologically-scoped BroadCast:** Topologically-Scoped Broadcast (TSB) es un tipo de comunicación en el que un mensaje se difunde a los nodos cercanos, pero con un límite en el número de saltos que puede dar. En lugar de definir una zona geográfica, como hace el GeoBroadcast, aquí se define cuántos “saltos” puede recorrer el mensaje desde el emisor (por ejemplo, 2 o 3 nodos de distancia). Así, el mensaje se propaga solo dentro de un radio limitado de nodos, lo que permite controlar el alcance y evitar que se envíe demasiado lejos. Este tipo de difusión es útil para enviar información local sin saturar toda la red.

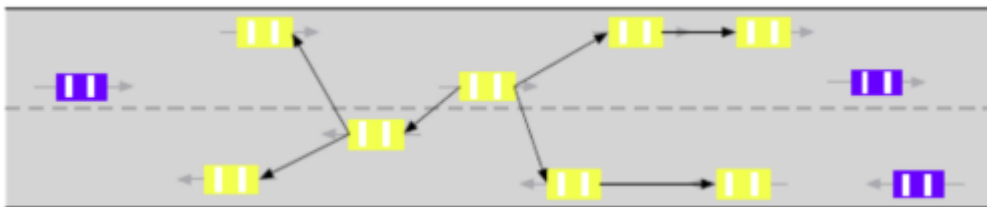


Figura 12. Representación de topologically-scoped broadcast

#### 4.4 Capa de Servicios (ITS Facilities Layer)

La capa de facilities actúa como intermediaria entre las capas de red/transporte y las aplicaciones ITS. Su función principal es ofrecer servicios comunes que simplifican el desarrollo de aplicaciones cooperativas y garantizan la interoperabilidad.

- **Gestión y agregación de datos.**
- **Procesamiento de mensajes.**
- **Direccionamiento de datos a aplicaciones y gestión de sesiones de comunicación.**
- **Administración de servicios ITS**, como el descubrimiento y actualización de módulos software.

Gracias a esta capa, las aplicaciones pueden operar de forma abstracta, sin preocuparse por los detalles técnicos de la comunicación subyacente, lo que facilita la escalabilidad y compatibilidad entre sistemas.

##### 4.4.1 ETSI common data dictionary (CDD)

El Diccionario Común de Datos ITS (CDD) es un repositorio de objetos de datos llamados Data Elements (DEs) y Data Frames (DFs) que representan datos, así como la información necesaria para la implementación de aplicaciones ITS y funcionalidades ITS. Un DE/DF puede ser utilizado para construir mensajes de la capa de facilidades ITS o de la capa de aplicaciones ITS, si el mensaje en cuestión lo requiere. Cada objeto de datos está definido por un conjunto de atributos, que permiten identificar los datos en cuestión. Estos atributos están definidos en la documentación oficial de ETSI, concretamente ETSI TS 102 894-2. Estos objetos de datos constituyen la base estructural sobre la cual se construyen los distintos tipos de mensajes cooperativos, permitiendo una definición coherente, reutilizable y estandarizada de la información intercambiada entre entidades ITS.

- Data Element (DE): tipo de dato que contiene un solo dato (suelen ser valores primitivos).
- Data Frame (DF): tipo de dato que contiene más de un elemento de datos en un orden predefinido.

Según el propósito de uso, un DE o DF puede clasificarse en las siguientes categorías:

- Gestión de mensajes: el DE/DF se utiliza para apoyar la gestión de un mensaje de la capa de facilidades ITS o de la capa de aplicaciones ITS, así como del protocolo de comunicación, por ejemplo, la versión del protocolo.
- Uso en aplicaciones: el DE/DF incluye información y datos que son útiles para la implementación de una o varias aplicaciones ITS.

Para la definición formal de los Data Elements (DE) y Data Frames (DF), el estándar ETSI emplea la notación ASN.1 (Abstract Syntax Notation One). ASN.1 es un estándar internacional utilizado para describir estructuras de datos de manera abstracta, independiente del lenguaje de programación o del formato de codificación subyacente. Su objetivo principal es garantizar la interoperabilidad entre diferentes sistemas que intercambian información estructurada.

ASN.1 permite especificar con precisión cómo deben interpretarse y codificarse los datos transmitidos entre entidades cooperativas (vehículos, infraestructura, etc.), asegurando una comprensión común y sin ambigüedades.

**Ejemplo de un DE definido por ETSI**

```

/**
 * This DE indicates the acceleration confidence value which
 * represents the estimated absolute accuracy of an acceleration value
 * with a default confidence level of 95 %.
 * If required, the confidence level can be defined by the
 * corresponding standards applying this DE.
 *
 * The value shall be set to:
 * - `n` (`n > 0` and `n < 101`) if the confidence value is equal to
 * or less than  $n \times 0,1 \text{ m/s}^2$ , and greater than  $(n-1) \times 0,1 \text{ m/s}^2$ ,
 * - `101` if the confidence value is out of range i.e. greater than
 * 10  $\text{m/s}^2$ ,
 * - `102` if the confidence value is unavailable.
 *
 * The value 0 shall not be used.
 *
 * @note: The fact that an acceleration value is received with
 * confidence value set to `unavailable(102)` can be caused by several
 * reasons, such as:
 * - the sensor cannot deliver the accuracy at the defined confidence
 * level because it is a low-end sensor,
 * - the sensor cannot calculate the accuracy due to lack of variables,
 * or
 * - there has been a vehicle bus (e.g. CAN bus) error.
 * In all 3 cases above, the acceleration value may be valid and used
 * by the application.
 *
 * @note: If an acceleration value is received and its confidence
 * value is set to `outOfRange(101)`, it means that the value is not
 * valid and therefore cannot be trusted. Such value is not useful for
 * the application.
 *
 * @unit 0,1  $\text{m/s}^2$ 
 * @category: Kinematic information
 * @revision: Description revised in V2.1.1
 */

AccelerationConfidence ::= INTEGER {
    outOfRange          (101),
    unavailable         (102)
} (0..102)

```

- **AccelerationConfidence:** es el nombre del Data Element.
- **Tipo de dato:** INTEGER
- **Rango:** (0..102): se especifica el rango completo de valores permitidos, desde 0 hasta 102. Esto permite definir la cardinalidad del DE y verificar su validez durante la codificación o validación.
- **Etiquetas simbólicas:**
  - outOfRange (101): indica que el valor de aceleración supera el rango medible por el sensor o sistema.
  - unavailable (102): indica que no hay una medida de aceleración disponible (por ejemplo, si el sensor está desactivado o hay un error de lectura).

**Ejemplo de un DF definido por ETSI**

```

/**
 * This DF represents activation data for real-time systems designed
 * for operations control, traffic light priorities, track switches,
 * barriers, etc.
 * using a range of activation devices equipped in public transport
 * vehicles.
 *
 * The activation of the corresponding equipment is triggered by the
 * approach or passage of a public transport
 * vehicle at a certain point (e.g. a beacon).
 *
 * @field ptActivationType: type of activation.
 *
 * @field ptActicationData: data of activation.
 *
 * Today there are different payload variants defined for public
 * transport activation-data. The R09.x is one of
 * the industry standard used by public transport vehicles (e.g.
 * buses, trams) in Europe (e.g. Germany Austria)
 * for controlling traffic lights, barriers, bollards, etc. This DF
 * shall include information like route, course,
 * destination, priority, etc.
 *
 * The R09.x content is defined in VDV recommendation 420 [7]. It
 * includes following information:
 * - Priority Request Information (pre-request, request, ready to
 * start)
 * - End of Prioritization procedure
 * - Priority request direction
 * - Public Transport line number
 * - Priority of public transport
 * - Route line identifier of the public transport
 * - Route number identification
 * - Destination of public transport vehicle
 *
 * Other countries may use different message sets defined by the
 * local administration.
 * @category: Vehicle information
 * @revision: V1.3.1
 */

PtActivation ::= SEQUENCE {
    ptActivationType    PtActivationType,
    ptActivationData    PtActivationData
}

```

- **PtActivation:** es el nombre del Data Frame. En este caso, se refiere a la activación de una función relacionada con el transporte público (Public Transport).
- **Tipo de estructura:** SEQUENCE: indica que los elementos que componen el DF deben aparecer en el orden exacto en que han sido definidos. Todos los campos especificados son obligatorios, salvo que alguno de ellos esté marcado explícitamente como OPTIONAL (no es el caso aquí).



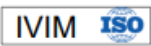
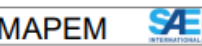
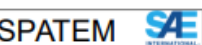



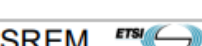
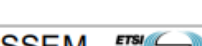
**Atributos del DF:** ptActivationType y ptActivationData.

#### 4.4.2 Tipos de mensajes

Los tipos de mensajes ITS son la forma estructurada en la que se transmite la información cooperativa entre estaciones ITS (vehículos, infraestructuras, centros de control...). Estos mensajes se definen siguiendo las estructuras de datos normalizadas en el ETSI Common Data Dictionary (CDD), mediante Data Elements (DEs) y Data Frames (DFs), tal como se ha explicado en la sección anterior. Entre los mensajes estandarizados más relevantes y utilizados en la capa de facilities, destacan:

- **CAM (Cooperative Awareness Message):** es un mensaje periódico transmitido por vehículos y otras estaciones ITS para informar de su presencia, posición, estado dinámico y atributos básicos a los demás usuarios de la vía. Su propósito es crear y mantener la conciencia cooperativa en el entorno V2X, permitiendo que vehículos e infraestructuras conozcan la ubicación y el movimiento de los demás en tiempo real, lo que respalda múltiples aplicaciones de seguridad vial.
- **DENM (Decentralized Environmental Notification Message):** es un mensaje event-driven (activado por eventos) utilizado para alertar a las estaciones ITS cercanas sobre peligros viales o condiciones anómalas detectadas, como accidentes, obstáculos súbitos, climatología adversa o incidencias de tráfico inesperadas. El mensaje incluye información detallada del evento (tipo de peligro, posición, hora, etc.) y se difunde de manera descentralizada a través de la red ITS para que los conductores en las proximidades puedan tomar precauciones o acciones preventivas.
- **IVIM (In-Vehicle Information Message):** es un mensaje emitido desde la infraestructura vial hacia los vehículos, que proporciona información de señales de tráfico y regulaciones viales (tanto estáticas como dinámicas) directamente a los sistemas a bordo del vehículo. Este mensaje puede incluir datos sobre límites de velocidad contextuales, restricciones de paso (por peso, tamaño, emisiones, etc.), avisos de obras en la vía u otras indicaciones obligatorias o recomendadas, reflejando así en el vehículo las señales físicas de la carretera (o señales virtuales definidas para C-ITS). IVIM permite que los conductores y vehículos reciban en tiempo real las normas viales relevantes del tramo, mejorando el cumplimiento y la seguridad.
- **MAPEM (MAP Extended Message):** es un mensaje emitido típicamente por las unidades de carretera (RSU) hacia los vehículos, que contiene una descripción digital detallada de la topología vial local, usualmente la geometría completa de una intersección o tramo de carretera. A través del MAPEM se transmiten datos estáticos de la infraestructura, como la configuración de carriles y sus conexiones, la ubicación de líneas de detención, identificadores de intersección y las maniobras permitidas en cada aproximación. En esencia, provee un mapa electrónico cooperativo que sirve de referencia para que los vehículos interpreten correctamente la información de semáforos (SPAT) u otros mensajes en entornos viarios complejos.
- **SPATEM (Signal Phase and Timing Extended Message):** es un mensaje de infraestructura a vehículo diseñado para comunicar en tiempo real el estado actual y los cambios próximos de un cruce semaforizado a los vehículos que se aproximan. Un SPATEM típico incluye la fase de semáforo vigente para cada grupo de señales (es decir, qué luces están en verde, ámbar o rojo) e indica el tiempo restante para la siguiente conmutación de estado, y además puede incorporar datos como recomendaciones de velocidad para lograr una ola verde, información sobre la longitud de colas de vehículos o la indicación de si hay alguna solicitud de prioridad activa en el sistema.

- **CPM (Collective Perception Message):** es un mensaje mediante el cual una estación ITS (ya sea un vehículo o un sensor en la infraestructura) comparte con las estaciones cercanas información sobre los objetos que percibe en su entorno local, ampliando la conciencia situacional colectiva. Este mensaje contiene una lista de objetos detectados – como vehículos, peatones, ciclistas, animales u otros obstáculos relevantes – junto con sus características observadas (posición, velocidad, dirección e incluso tamaño estimado), y puede incluir también zonas del camino detectadas como libres u ocupadas. Intercambiando CPM, los vehículos pueden “ver” más allá de los límites de sus propios sensores, detectando usuarios de la vía que no estén equipados con V2X y fusionando datos de múltiples fuentes para reducir la incertidumbre en la percepción del entorno.
- **MCM (Maneuver Coordination Message):** es un mensaje diseñado para intercambiar planes de trayectoria futuros entre vehículos conectados (especialmente vehículos automatizados), con el fin de coordinar maniobras cooperativas en la carretera. A través del MCM, un vehículo puede comunicar a los circundantes su trayectoria prevista (por ejemplo, notificar con antelación un cambio de carril, una incorporación o maniobra de adelantamiento planificada), de modo que los demás vehículos puedan ajustar su comportamiento y negociar movimientos de forma segura y eficiente de manera distribuida. Esta capacidad de compartir intenciones futuras permite una planificación de trayectorias conjunta, mejorando la seguridad y la fluidez del tráfico en escenarios de tráfico denso o maniobras complejas.
- **VAM (VRU Awareness Message):** es un mensaje de conciencia cooperativa transmitido periódicamente por los VRU (Usuarios Vulnerables de la Vía, como peatones, ciclistas o motociclistas) equipados con dispositivos ITS, para notificar su presencia, posición y movimiento a los vehículos e infraestructuras cercanas. Funciona de manera análoga a la CAM de los vehículos, extendiendo la conciencia cooperativa del sistema de transporte a estos usuarios más vulnerables y de comportamiento más impredecible. El VAM típicamente incluye datos cinemáticos básicos del VRU (posición, velocidad, direcciones actuales) y puede incorporar además parámetros de predicción de movimiento para estimar la trayectoria futura a corto plazo del usuario.
- **SREM (Signal Request Extended Message):** es un mensaje de vehículo a infraestructura mediante el cual vehículos autorizados (por ejemplo, autobuses de transporte público o vehículos de emergencia) solicitan prioridad en un semáforo próximo. A través de un SREM, el vehículo comunica al controlador de tráfico su petición de dar preferencia (prioridad verde) o preempción en la intersección, incluyendo identificadores y parámetros relevantes que describen la solicitud. Esta funcionalidad permite implementar medidas de prioridad semafórica en entornos C-ITS, como dar paso preferente a un tranvía o prolongar la luz verde para una ambulancia que se aproxima, mejorando la eficiencia y seguridad.
- **SSEM (Signal Status Extended Message):** es el mensaje de respuesta enviado desde la infraestructura (el controlador del semáforo) hacia el vehículo que previamente envió un SREM, e informa sobre el resultado de su solicitud de prioridad. A través del SSEM, el sistema de control de tráfico notifica si la petición ha sido concedida, rechazada (cancelada) o si se ha modificado algún aspecto de esta, proporcionando confirmación y detalles al solicitante. De este modo, el SSEM cierra el bucle de comunicación en la gestión cooperativa de semáforos, permitiendo al vehículo conocer el estado de su solicitud y adaptar su conducción en consecuencia (por ejemplo, prepararse para detenerse si la prioridad no fue otorgada).

	CAM	Cooperative Awareness Message - Location and dynamics of the Vehicle/RSU/Pedestrian
	DENM	Decentralized Environmental Notification Message - Road Hazards / Events
	IVIM	In-Vehicle Information Message - Road Information
	MAPEM	MAP (topology) Extended Message - Road layout
	SPATEM	Signal Phase And Timing Extended Message - Traffic Light Phase and Timing
	CPM	Collective Perception Message – Ego vehicle perception information
	MCM	Maneuver Coordination Message - Manouvre orders and coordination between stations
	VAM	VRU Awareness Message - Location and dynamics of the VRUs
	SREM	Signal Request Extended Message - Priority request
	SSEM	Signal Request Extended Message - Priority status

**Figura 13.** Tipos de mensajes definidos por las entidades estandarizadoras (ETSI, SAE y ISO)

#### 4.5 Capa de Aplicación (ITS Applications Layer)

La capa de aplicación representa el nivel más alto en la arquitectura de una estación ITS. Estas aplicaciones hacen uso directo de los mensajes definidos en la capa de facilities y pueden operar de forma autónoma o coordinada, dependiendo del tipo de servicio. A continuación, se describen algunas de las principales aplicaciones cooperativas C-ITS y los tipos de mensajes que emplean:

- **PVD (Probe Vehicle Data):** es una aplicación que permite recolectar información dinámica de los vehículos en circulación para su análisis posterior en sistemas de gestión del tráfico o mantenimiento predictivo. Utiliza mensajes CAM y CPM para transmitir datos como posición, velocidad, aceleración, condiciones del entorno detectadas por sensores (por ejemplo, lluvia, niebla) u objetos próximos, convirtiendo cada vehículo en una “sonda móvil” que contribuye a una visión colectiva del estado de la red vial.
- **RHW (Road Hazard Warning):** tiene como objetivo advertir a los conductores de la existencia de peligros específicos en la vía, como vehículos averiados, objetos en la calzada, condiciones meteorológicas extremas o accidentes recientes. Esta aplicación se basa en el mensaje DENM, que permite generar alertas descentralizadas con información contextual del evento (ubicación, tipo de peligro, tiempo de validez) y difundirlas a otros usuarios en tiempo real.

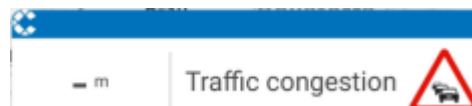


Figura 14. Ejemplo de aviso

- **GLOSA (Green Light Optimal Speed Advisory):** proporciona a los conductores una recomendación de velocidad óptima para alcanzar una intersección semafórica durante la fase verde, evitando así detenciones innecesarias y reduciendo el consumo de combustible. Esta funcionalidad requiere información detallada de la geometría de la intersección, obtenida mediante el mensaje MAPEM, y de la temporización semafórica, proporcionada por el mensaje SPATEM. Combinando ambos, GLOSA calcula y comunica al conductor la mejor velocidad de aproximación.

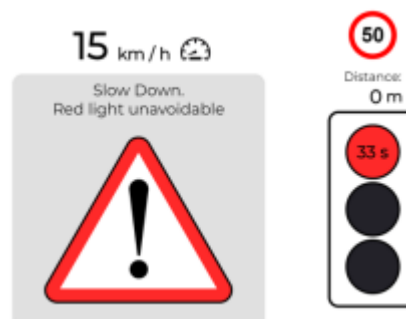


Figura 15. Ejemplo de interfaz GLOSA

- **IVS (In-Vehicle Signage):** ofrece al conductor una representación visual de señales de tráfico dentro del vehículo, lo que resulta especialmente útil en situaciones donde la señalización física puede ser confusa, inexistente o difícil de percibir. Esta aplicación utiliza el mensaje IVIM, definido por ISO, para transmitir información como límites de velocidad temporales, carriles cerrados, desvíos o zonas de obras.



Figura 16. Ejemplo de interfaz in-vehicle alerts

- **EVW (Emergency Vehicle Warning):** permite alertar a los vehículos cercanos sobre la aproximación de un vehículo de emergencia, incluso cuando aún no es visible ni audible. Dependiendo del modo de implementación, esta aplicación puede utilizar mensajes CAM, cuando el vehículo de emergencia transmite su posición de forma continua, o DENM, si emite una alerta puntual indicando su prioridad de paso o trayecto.



Figura 17. Ejemplo de aviso (EVW)

- **EEBL (Electronic Emergency Brake Light):** es una aplicación diseñada para alertar a los conductores de vehículos que circulan por detrás sobre una frenada brusca del vehículo precedente, reduciendo así el riesgo de colisiones por alcance. El aviso se transmite mediante un mensaje DENM generado automáticamente cuando el sistema detecta una deceleración de emergencia, lo que permite a los vehículos posteriores reaccionar anticipadamente.
- **GP (Green Priority):** está orientada a otorgar prioridad de paso a ciertos vehículos (por ejemplo, ambulancias, autobuses o servicios logísticos) en intersecciones semafóricas. Para ello, el vehículo emite un mensaje SREM (Service Request Message) solicitando la prioridad, y la infraestructura responde con un SSEM (Service Status Message), confirmando si la solicitud ha sido aceptada y cuándo se aplicará la fase favorable. Esta aplicación permite reducir los tiempos de respuesta de los servicios de emergencia y mejorar la eficiencia del transporte público.

## 5 Implementación de la plataforma ITS de IDIADA

### 5.1 Introducción a la plataforma ITS de IDIADA

La empresa Applus+ IDIADA, con sede en Tarragona, está desarrollando una plataforma ITS propia basada en comunicaciones vehiculares. Durante mi periodo de prácticas (3 meses) —integrado en la realización de este Trabajo Fin de Grado— he tenido la oportunidad de colaborar directamente en el diseño e implementación de dicha plataforma.

Applus+ IDIADA ofrece servicios de diseño, ingeniería y homologación para el sector de la automoción. Gracias a ello, dispone de pistas de ensayo propias y varias RSU instaladas, lo que permite realizar pruebas en entornos controlados para validar distintos casos de uso V2X.

Parte de la plataforma ITS de Applus+ IDIADA se despliega en AWS y expone una API REST que permite a clientes externos —incluso desarrollados en otros lenguajes o frameworks— conectarse y enviar sus propios mensajes ITS, siempre que cumplan la estructura y el formato ASN.1 definidos por ETSI, permitiendo así interoperabilidad. Está desarrollada principalmente en Java. La elección de Java es porque las librerías de los codificadores de mensajes ASN1 que Applus+ IDIADA usa están compiladas en Java.

En esta sección 5 se ofrece una descripción global de la arquitectura de la plataforma ITS desarrollada por Applus+ IDIADA, detallando sus principales componentes, módulos y flujos de información. Esta visión de conjunto servirá de marco de referencia para entender cómo interactúan entre sí los diferentes elementos —desde las unidades a bordo de los vehículos (OBU) y las estaciones de carretera (RSU), hasta los servicios centrales en la nube—, así como los protocolos y estándares que garantizan la interoperabilidad y la seguridad del sistema.

Todo el contenido técnico que se expone a continuación corresponde al diseño e implementación original de Applus+ IDIADA; mi trabajo se ha centrado en integrar y optimizar determinadas funcionalidades dentro de ese entorno ya establecido. Durante mi estancia de prácticas, he colaborado estrechamente con el equipo de desarrollo siguiendo un marco ágil Scrum: las diferentes actividades se planificaron en sprints de 2-3 semanas, con reuniones diarias de seguimiento (daily stand-up), revisiones de sprint y retrospectivas.

Gracias a esta organización, realicé tareas —que se describen en detalle en la sección 6— relacionadas con la codificación de nuevos Data Frames, la adaptación de librerías ASN.1, la creación de tests automatizados y la configuración de entornos de despliegue. Cada tarea formó parte de un sprint y se priorizó en función de su impacto y urgencia, y se completó dentro de los plazos marcados por el diagrama de Gantt que acompaña al planning general. De este modo, mi contribución no solo reforzó mis competencias técnicas —en Java, Docker, Kubernetes, MQTT y herramientas de integración continua—, sino que también me permitió vivir de primera mano el ciclo completo de desarrollo de software en un entorno profesional real.

### 5.1.1 *Objetivos*

- Estudio y análisis de la plataforma ITS ya existente de IDIADA y realizar documentación sobre ella.
- Realización de tareas que contribuyan al desarrollo continuo de esta plataforma.
- Aprendizaje continuo sobre nuevas herramientas de desarrollo.
- Aprendizaje profundo sobre tecnologías ya existentes que se han aprendido a lo largo de la carrera (Java con nuevos patrones de diseños no implementados).
- Fomentar el trabajo colaborativo dentro del equipo y reforzar habilidades de comunicación y coordinación.

### 5.1.2 *Listado de tareas realizadas*

- **Clean and update Maven libraries in several projects:** Revisar y ajustar las dependencias de Maven en distintos proyectos, eliminar versiones obsoletas y garantizar que las librerías necesarias queden correctamente clasificadas entre los ámbitos de compilación y prueba (test).
- **Implement AOP in MessageExtractor:** Aplicando Programación orientada a aspectos (AOP), diseñar e integrar aspectos para centralizar el logging en el servicio de MessageExtractor.
- **Investigate CAM compatibility between camV2 and camRel21:** investigar porque no hay retrocompatibilidad entre versión V2 y Rel21 del CAM, cuando solía haber entre diferentes versiones de CAM.
- **CDDv4 implementation:** Desarrollar y validar las nuevas reglas y estructuras de datos definidas en la versión 4 del diccionario de datos comunes de ETSI.
- **Get Asn1Type from DF in adaptationLib new CAM version:** Adaptar la librería adaptationLib para leer el nuevo DF (DataFrame) del CAM (Rel22) y mapearlas al tipo asn1Type definido por ETSI.

Los detalles de cómo se han implementado estas tareas se encuentran en la sección 6.

### 5.1.3 *Diagrama de Gantt*

A continuación, se muestra la planificación de las tareas de este TFG junto con su diagrama de Gantt.

#### 1. Investigación ITS/C-ITS

Duración: 11 days [Inicio: Wed 12/02/25 - Fin: Thu 27/02/25]

- 1.1. Investigación de sistemas ITS (1 day · Wed 12/02/25 – Thu 13/02/25)
- 1.2. Investigación de los sistemas C-ITS (3 days · Thu 13/02/25 – Tue 18/02/25)
  - 1.2.1. Componentes que forman parte de un sistema C-ITS (2 days · Thu 13/02/25 – Mon 17/02/25)
  - 1.2.2. Entidades encargadas de la estandarización (1 day · Mon 17/02/25 – Tue 18/02/25)
  - 1.2.3. Beneficios de los sistemas C-ITS (1 day · Mon 17/02/25 – Tue 18/02/25)

## Implementación de la plataforma ITS de IDIADA

- 1.3. Arquitectura de una plataforma ITS (7 days · Tue 18/02/25 – Thu 27/02/25)
  - 1.3.1. Capa de acceso (DSRC vs C-V2X) (4 days · Tue 18/02/25 – Mon 24/02/25)
  - 1.3.2. Capa de red y transporte (2 days · Mon 24/02/25 – Wed 26/02/25)
  - 1.3.3. Capa de servicios (1 day · Wed 26/02/25 – Thu 27/02/25)
  - 1.3.4. Capa de aplicación (1 day · Wed 26/02/25 – Thu 27/02/25)

### **2. Estudio de la plataforma ITS de IDIADA**

Duración: 44 days [Inicio: Tue 20/02/25 – Fin: Thu 23/04/25]

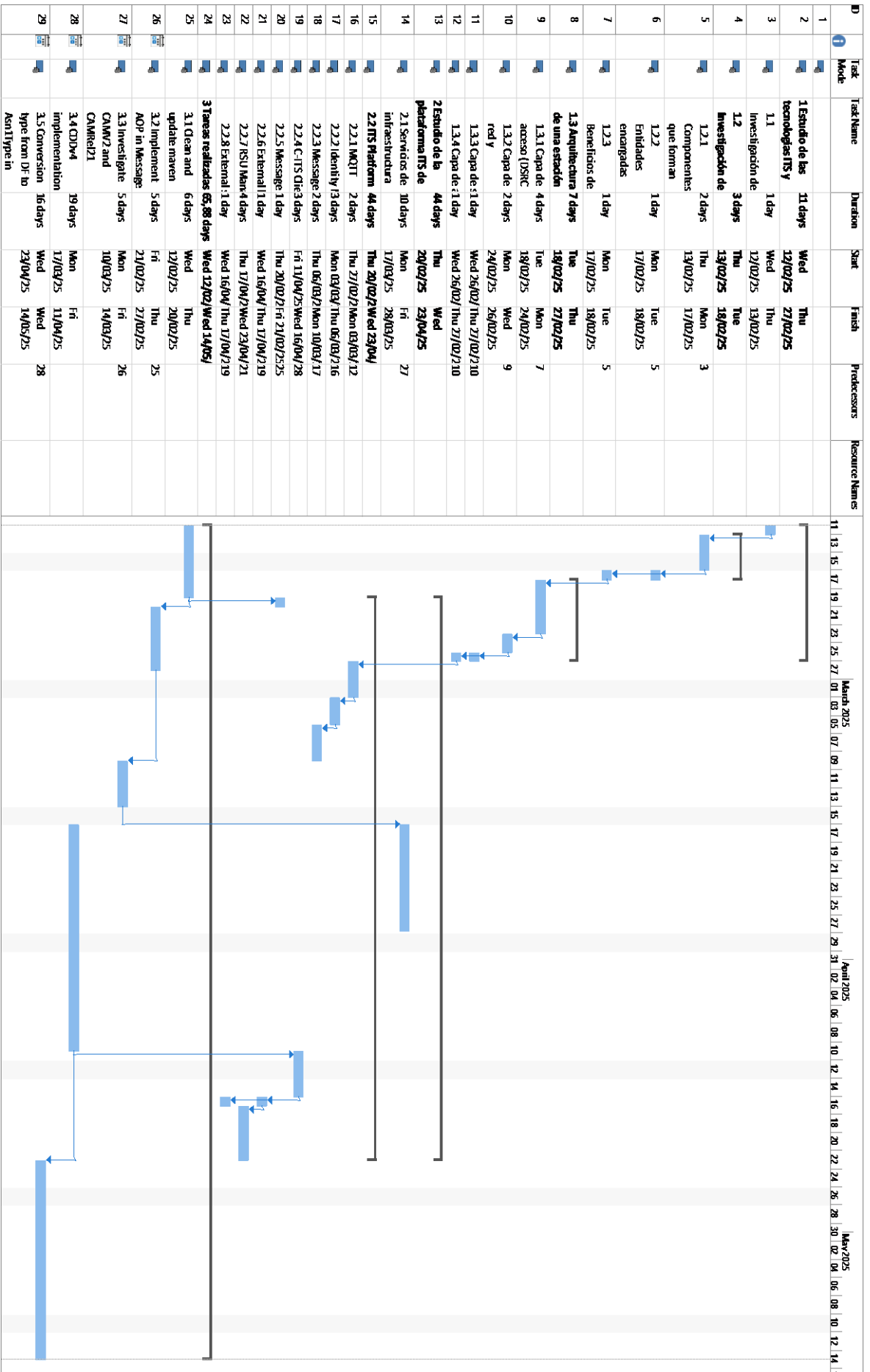
- 2.1. Servicios de infraestructura (10 days · Mon 17/03/25 – Fri 28/03/25)
- 2.2. ITS Platform (44 days · Thu 20/02/25 – Wed 23/04/25)
  - 2.2.1. MQTT (2 days · Thu 27/02/25 – Mon 03/03/25)
  - 2.2.2. Identity Manager (3 days · Mon 03/03/25 – Thu 06/03/25)
  - 2.2.3. Message Manager (2 days · Thu 06/03/25 – Mon 10/03/25)
  - 2.2.4. C-ITS Client (3 days · Fri 11/04/25 – Wed 16/04/25)
  - 2.2.5. Message Extractor (1 day · Thu 20/02/25 – Fri 21/02/25)
  - 2.2.6. External broker (1 day · Wed 16/04/25 – Thu 17/04/25)
  - 2.2.7. RSU Manager (4 days · Thu 17/04/25 – Wed 23/04/25)
  - 2.2.8. External services (1 day · Wed 16/04/25 – Thu 17/04/25)

### **3. Tareas realizadas**

Duración: 66 days [Inicio: Wed 12/02/25 – Fin: Wed 14/05/25]

- 3.1. Clean and update maven libraries (6 days · Wed 12/02/25 – Thu 20/02/25)
- 3.2. Implement AOP in Message Extractor (5 days · Fri 21/02/25 – Thu 27/02/25)
- 3.3. Investigate CAMV2 and CAMRel21 compatibility (5 days · Mon 10/03/25 – Fri 14/03/25)
- 3.4. CDDv4 implementation (19 days · Mon 17/03/25 – Fri 11/04/25)
- 3.5. Conversion type from DF to Asn1Type en AdaptationLib (16 days · Wed 23/04/25 – Wed 14/05/25)

# Implementación de la plataforma ITS de IDIADA



## 5.2 Árbol de dependencias

Antes de abordar el diseño de la plataforma ITS, es necesario establecer previamente una serie de dependencias fundamentales que garanticen su correcto funcionamiento. Estas dependencias están relacionadas con la infraestructura de base, y abarcan la definición e implementación de los tipos de datos y las interfaces que se utilizarán en la plataforma. En concreto, se requiere disponer del CDD (Common Data Dictionary) con sus correspondientes Data Elements y Data Frames, así como la implementación de los diferentes mensajes estandarizados que compondrán la comunicación cooperativa dentro del sistema.

Una vez definidos estos elementos de base, es necesario contar con una librería de adaptación que hará de puente entre la infraestructura y los servicios. Al final, cualquier servicio de la plataforma ITS importará esta librería como una dependencia más del proyecto y así ser capaz de gestionar el formato de los mensajes cooperativos, tanto en su recepción como en su envío.

**[INFORMACIÓN CONFIDENCIAL]**

**[FIN INFORMACIÓN CONFIDENCIAL]**

### **5.3 ETSI common data dictionary (CDD)**

**[INFORMACIÓN CONFIDENCIAL]**

**[FIN INFORMACIÓN CONFIDENCIAL]**

**5.3.1 Data Elements**

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

5.3.1.1 Ejemplo de un DE del tipo enumeración.

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

### 5.3.2 Data Frames

En el estándar ETSI, los Data Frames (DF) pueden estructurarse principalmente de dos formas, según la notación ASN.1: mediante **SEQUENCE** o mediante **CHOICE**. Estas estructuras determinan cómo deben organizarse y validarse los atributos que componen un DF.

- **SEQUENCE**: representa una estructura ordenada de campos, en la que los atributos deben definirse y construirse siguiendo estrictamente el orden especificado en la norma. Es la forma más común de estructurar un DF, ya que permite una construcción lineal, clara y determinista. Dentro de una SEQUENCE, es posible que algunos campos estén marcados con el modificador OPTIONAL. Este flag indica que el campo correspondiente no es obligatorio y puede omitirse si no se dispone del dato o si no es aplicable en el contexto. Sin embargo, si se incluye, debe cumplir con el tipo y restricciones definidos.

#### Ejemplo de un DF Sequence de ETSI ASN1

```
AccelerationPolarWithZ ::= SEQUENCE {
    accelerationMagnitude           AccelerationMagnitude,
    accelerationDirection           CartesianAngle,
    zAcceleration                   AccelerationComponent
OPTIONAL
```

- **CHOICE**: esta estructura se utiliza cuando un mismo DF puede adoptar diferentes formas, permitiendo que solo uno de los atributos definidos esté presente en una instancia determinada. Los demás campos deben estar ausentes, nulos o explícitamente vacíos. Esto obliga a implementar lógica de validación adicional en la construcción del objeto, para garantizar que solo un campo esté activo en cada momento. Es especialmente útil en situaciones donde el mensaje puede contener distintos tipos de información según el caso de uso.

#### Ejemplo de un DF CHOICE de ETSI ASN1

```
MapReference ::= CHOICE {
    roadsegment      RoadSegmentReferenceId,
    intersection     IntersectionReferenceId
}
```

Además, ciertos Data Frames pueden contener atributos sujetos a restricciones adicionales, como rangos numéricos válidos, relaciones de dependencia entre campos, o condiciones lógicas específicas. Estas restricciones deben aplicarse durante la construcción de los objetos, para garantizar que cualquier instancia generada sea válida y conforme al estándar ETSI.

#### Ejemplo de un DF con ciertas restricciones

```
RadialShapeDetails ::= SEQUENCE {
    range                               StandardLength12b,
    horizontalOpeningAngleStart         CartesianAngleValue,
    horizontalOpeningAngleEnd           CartesianAngleValue,
    verticalOpeningAngleStart           CartesianAngleValue OPTIONAL,
    verticalOpeningAngleEnd             CartesianAngleValue OPTIONAL
}
((WITH COMPONENTS {..., verticalOpeningAngleStart ABSENT,
verticalOpeningAngleEnd ABSENT }) |
(WITH COMPONENTS {..., verticalOpeningAngleStart PRESENT,
verticalOpeningAngleEnd PRESENT })))
```

Los campos `verticalOpeningAngleStart` y `verticalOpeningAngleEnd` son opcionales, pero no pueden usarse de forma independiente. La cláusula `WITH COMPONENTS` impone una restricción de consistencia: o bien ambos están ausentes, o bien ambos deben estar presentes. Esta condición asegura que, en caso de definir una apertura vertical, esta esté completamente especificada, evitando ambigüedades o configuraciones incompletas en la interpretación del DF.

### 5.3.2.1 Ejemplo de un DF del tipo SEQUENCE

**[INFORMACIÓN CONFIDENCIAL]**

**[FIN INFORMACIÓN CONFIDENCIAL]**

5.3.2.2 Ejemplo de un DF del tipo CHOICE

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

**Ejemplo de pruebas unitarias de builds en un DF CHOICE**

**[INFORMACIÓN CONFIDENCIAL]**

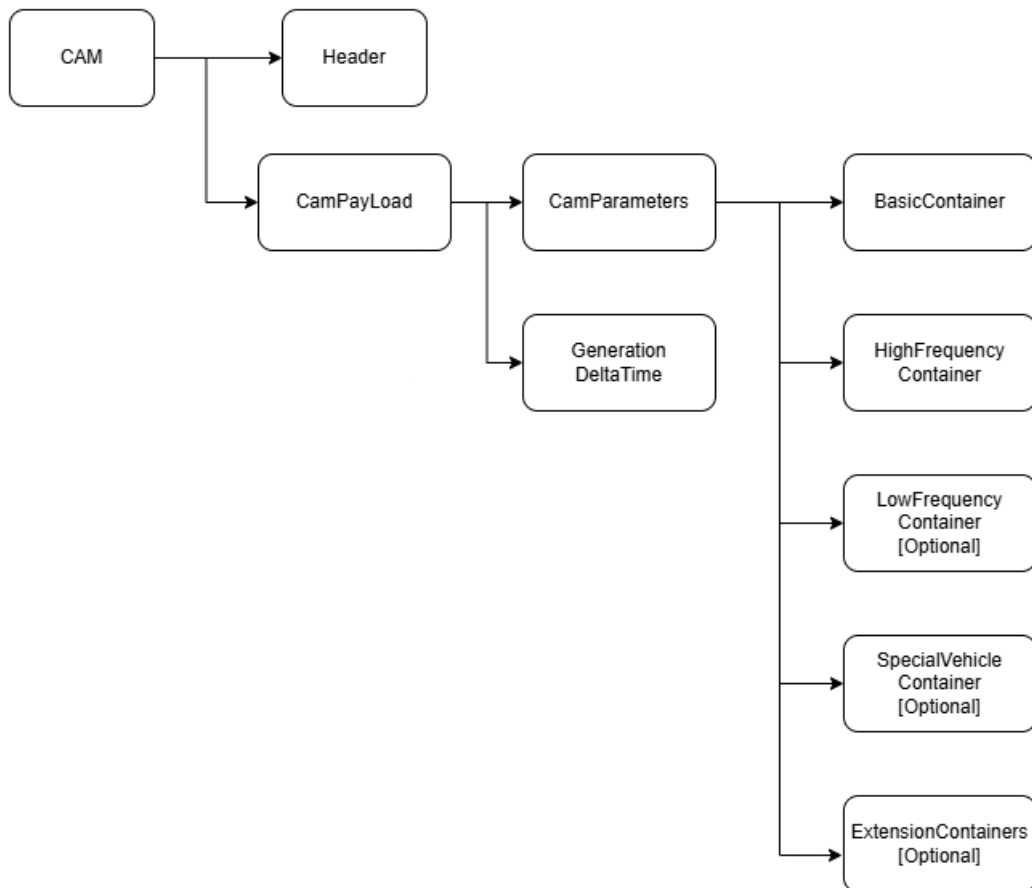
**[FIN INFORMACIÓN CONFIDENCIAL]**

## 5.4 Mensajes ITS

Para la implementación de los mensajes está dividido en 2 partes: la propia implementación del CAM y la parte de codificación y decodificación en formatos ASN.1 UPER (Unaligned Packed Encoding Rules). Este formato será usado para la transmisión de los mensajes.

La implementación propia de IDIADA de los mensajes ITS sigue la misma metodología estructurada utilizada en la definición de objetos en el ETSI Common Data Dictionary (CDD), concretamente de los DF. A modo ilustrativo, se presentará a continuación un ejemplo práctico centrado en uno de los mensajes más ampliamente utilizados en el ecosistema C-ITS: el CAM (Cooperative Awareness Message). Aunque en esta sección solo se detallará la implementación de este mensaje concreto, cabe destacar que el resto de los mensajes definidos en el marco C-ITS (como DENM, IVIM, SPATEM, etc.) siguen el mismo proceso de implementación, basado en la reutilización de estructuras normalizadas y en el uso de la notación ASN.1 para la definición de sus campos y restricciones.

### 5.4.1 Implementación del CAM



**Figura 18.** Diagrama simplificado de los atributos que lleva el CAM

Esta es la estructura general simplificada que tiene el CAM. Para ver más detalles, revisar el estándar publicado por ETSI, concretamente ETSI TS 103 900. Como dicho anteriormente, la implementación sigue el mismo procedimiento que los Data Frames, por lo que se mostrarán sólo porciones de código relevantes.

**[INFORMACIÓN CONFIDENCIAL]**

**[FIN INFORMACIÓN CONFIDENCIAL]**

Este proceso se repite para todos los DF definidos en el CAM.

**5.4.2 Implementación de codificadores y decodificadores**

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

**5.4.3 Adaptation-Lib**

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

## 5.5 ITS-Platform

La plataforma ITS de IDIADA se puede resumir con el siguiente diagrama:

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

**5.5.1 GeoMessaging**

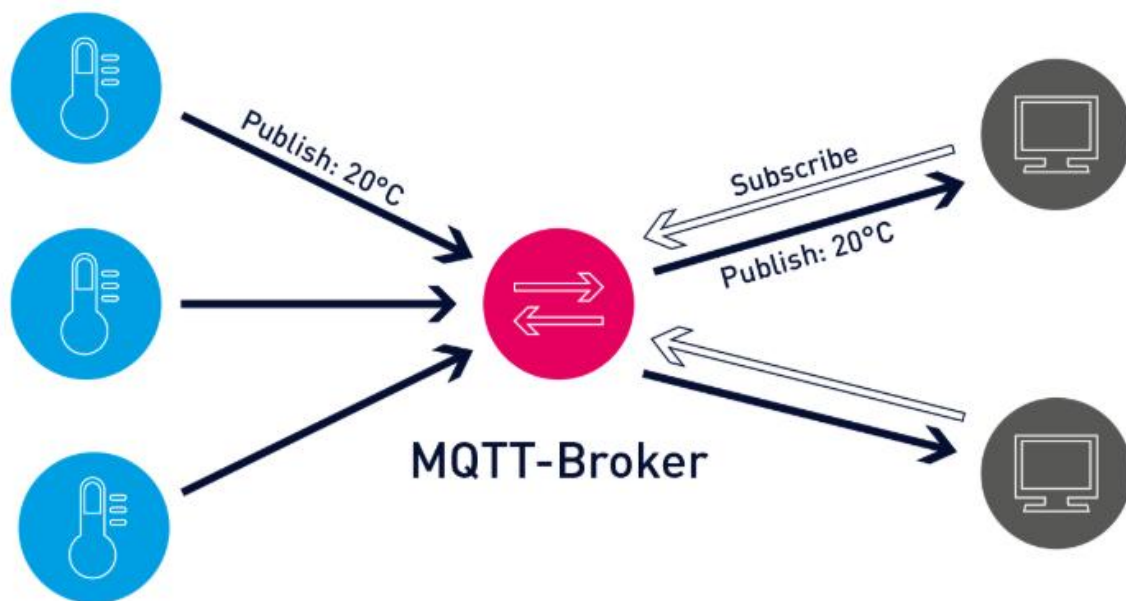
**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

### **5.5.2 Geomessaging internal broker (MQTT)**

MQTT (Message Queuing Telemetry Transport) es un protocolo de mensajería ligero y eficiente diseñado para comunicaciones en entornos con conectividad limitada. Originalmente desarrollado a finales de los años 90 para monitorizar oleoductos remotos, MQTT se ha convertido en un estándar ampliamente adoptado en el Internet de las Cosas (IoT). Destaca por su simplicidad y bajo peso, lo que le permite funcionar en dispositivos con recursos restringidos (CPU, memoria, batería) y en redes de bajo ancho de banda, alta latencia o incluso conexiones poco fiables. Su diseño se enfoca en minimizar el tráfico de red y la sobrecarga de comunicación manteniendo conexiones persistentes, facilitando así actualizaciones rápidas con baja latencia. En este contexto, MQTT resulta de interés para aplicaciones vehiculares cooperativas (C-ITS), donde numerosos vehículos y unidades en carretera deben intercambiar datos de forma confiable y en tiempo real bajo condiciones de red variables.

A diferencia de los modelos cliente-servidor tradicionales, MQTT emplea una arquitectura de publicación/suscripción (pub/sub). En este paradigma, los dispositivos actúan como editores (publicadores) o suscriptores, o ambos a la vez. Los editores no envían mensajes directamente entre sí, sino que publican sus datos en temas (topics) específicos en un servidor intermedio denominado broker (o corredor). Simultáneamente, los suscriptores se registran en aquellos temas de su interés y reciben del broker los mensajes pertinentes.

Esta intermediación desacopla completamente emisores y receptores: un publicador no necesita conocer qué clientes recibirán sus datos, y un suscriptor no necesita saber quién origina los mensajes. El broker MQTT se encarga de filtrar y enrutar los mensajes entrantes hacia todos los suscriptores de cada tema, gestionando además suscripciones (incluyendo comodines jerárquicos para agrupar múltiples temas) y controlando la distribución de datos. Esta arquitectura muchos-a-muchos proporciona gran flexibilidad y eficiencia en la difusión de información.



**Figura 19.** Diagrama representativo del patrón publish/subscribe

MQTT se ejecuta sobre la pila TCP/IP, aprovechando la confiabilidad del transporte TCP para entregar mensajes de forma segura y ordenada. Típicamente utiliza el puerto 1883 para conexiones TCP sin cifrar (y 8883 para conexiones seguras mediante TLS/SSL), estableciendo un enlace persistente entre cliente y broker.

### 5.5.2.1 Propiedades clave de MQTT que favorecen su uso en C-ITS

- **Ligereza y eficiencia:** Posee un diseño minimalista pensado para dispositivos embebidos e IoT. Sus mensajes son muy pequeños (encabezado fijo de 2 bytes) y el protocolo evita comunicaciones superfluas. Esto se traduce en mínima carga de red y bajo uso de energía, permitiendo operar sobre enlaces de capacidad limitada sin saturarlos.
- **Bajo consumo de ancho de banda:** Relacionado con lo anterior, MQTT minimiza la cantidad de datos intercambiados. Al ser un protocolo orientado a eventos (no sondeo constante), únicamente se transmiten mensajes cuando hay nuevos datos, reduciendo tráfico redundante.
- **Soporte de calidad de servicio (QoS):** MQTT ofrece tres niveles de QoS configurables para el envío de mensajes. Esto permite ajustar el equilibrio entre confiabilidad y overhead según la importancia de los datos:
  - **QoS 0** – “entrega al máximo una vez”: el mensaje se envía sin acuse de recibo ni reintentos (no garantiza llegada), minimizando la transmisión de datos.
  - **QoS 1** – “al menos una vez”: asegura que el mensaje llegue al menos una vez al destino, retransmitiéndolo hasta recibir confirmación (puede haber duplicados si fallan los ack).
  - **QoS 2** – “exactamente una vez”: garantiza la entrega una sola vez mediante un intercambio de cuatro pasos entre emisor y receptor, evitando duplicados. Estos niveles permiten adaptar MQTT a distintos requisitos: desde telemetrías donde se tolera la pérdida eventual (QoS0) hasta notificaciones críticas que requieren certeza absoluta (QoS2). En cualquier caso, la selección apropiada de QoS brinda flexibilidad para cumplir requerimientos de fiabilidad sin perder eficiencia.
- **Persistencia de mensajes:** MQTT soporta mecanismos para no perder datos incluso si un cliente se desconecta temporalmente. Si un cliente se ha suscrito con una sesión persistente (no volátil), el broker almacenará en cola los mensajes dirigidos a ese cliente (con QoS 1 o 2) que se publiquen mientras está offline. Al reconectarse, el cliente recibirá esos mensajes pendientes, garantizando continuidad informativa. Asimismo, MQTT ofrece la función de mensajes retenidos (retained messages): el broker puede conservar el último mensaje publicado en un tema determinado, de modo que cualquier nuevo suscriptor a ese tema reciba inmediatamente la información más reciente disponible. Estas características aportan tolerancia a desconexiones e inestabilidad de red, asegurando que los datos importantes no se pierdan.
- **Escalabilidad:** El modelo pub/sub centralizado en el broker hace que MQTT sea altamente escalable. La adición de nuevos dispositivos clientes no requiere modificar los emisores ni receptores existentes, solo registrar sus suscripciones en el broker. Un único broker puede manejar gran cantidad de publicaciones y suscripciones concurrentes; de hecho, MQTT está concebido para soportar miles de dispositivos comunicándose simultáneamente con requisitos mínimos de red.

[INFORMACIÓN CONFIDENCIAL]

[FIN INFORMACIÓN CONFIDENCIAL]

### 5.5.2.2 QuadTiles

El sistema de enrutamiento geográfico por quadtiles es una técnica de indexación espacial utilizada para dividir el mapa en regiones jerárquicas y facilitar operaciones como la localización, el direccionamiento de mensajes y la agregación geográfica.

El mapa se estructura como un quadtree (árbol cuaternario), en el que cada región (tile) se divide recursivamente en cuatro subregiones del mismo tamaño. En el nivel 0, el mundo entero se representa con un único tile. En el nivel 1, se divide en 4 tiles; en el nivel 2, en 16; en el nivel 3, en 64; y así sucesivamente, siguiendo una cuadrícula de  $2^n \times 2^n$  tiles por nivel. Cada tile representa una región geográfica con mayor precisión a medida que se incrementa el nivel de profundidad.

Cada región puede identificarse mediante una secuencia jerárquica de índices, donde cada número indica la subzona seleccionada dentro de su tile padre. Por ejemplo, la cadena "1/2/3/0/0/1" representa una región específica con zoom de nivel 6 del árbol cuaternario. A esta estructura también se le denomina quadtile path y nos sirve para usarlo como tópico de suscripción por región, en el que se va a hablar en el siguiente apartado.

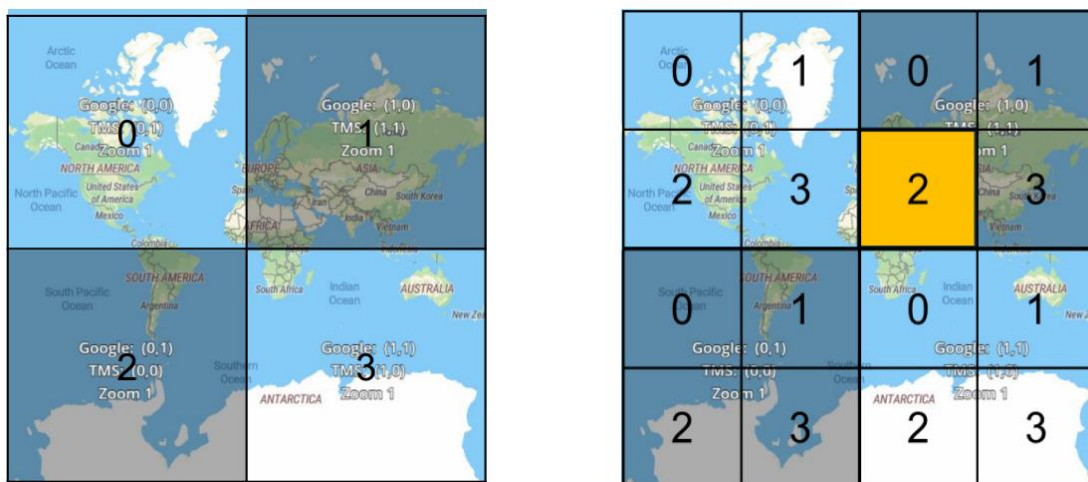


Figura 20. Representación del funcionamiento de QuadTiles

### 5.5.2.3 Funcionamiento de suscripción

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

**5.5.3 Identity Manager**

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

**5.5.4 Message Manager**

**[INFORMACIÓN CONFIDENCIAL]**

**[FIN INFORMACIÓN CONFIDENCIAL]**

**5.5.5 C-ITS Client**

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

**5.5.6 *Message Extractor***

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

**5.5.7 *External Broker***

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

**5.5.8 *RSU Manager***

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

**5.5.9 External services**

**[INFORMACIÓN CONFIDENCIAL]**  
**[FIN INFORMACIÓN CONFIDENCIAL]**

## **6 Tareas realizadas durante las prácticas**

### **6.1 Clean and update maven libraries in several projects**

**Fechas de la tarea:** 12/02/2025 – 20/02/2025

Esta tarea tuvo como objetivo principal identificar y corregir el uso incorrecto de dependencias en diversos proyectos Maven. Específicamente, se buscó detectar aquellas dependencias que deberían estar declaradas únicamente en el ámbito de pruebas (test) pero que estaban incluidas en el ámbito de compilación general (compile), lo cual puede conllevar costes innecesarios en entornos de producción, además de problemas de compatibilidad y mantenimiento.

**[INFORMACIÓN CONFIDENCIAL]**

**[FIN INFORMACIÓN CONFIDENCIAL]**

#### ***6.1.1 Conclusión de la tarea***

Esta tarea fue crítica para mejorar la estructura de dependencias de numerosos proyectos Maven dentro del entorno de desarrollo, reduciendo costes de ejecución, mejorando el mantenimiento y asegurando la compatibilidad entre versiones. Además, se sentaron buenas prácticas de uso del scope en Maven y se optimizó la trazabilidad de conflictos con herramientas especializadas como Dependency Analyzer y el árbol de dependencias.

Aquí solo se ha dejado reflejado 2 casos de uso particulares como ejemplo, pero se tuvo que realizar esta tarea para varios proyectos más.

## 6.2 Implement AOP in Message Extractor

**Fechas de la tarea:** 21/02/2025 – 27/02/2025

### 6.2.1 Introducción a AOP

Programación Orientada a Aspectos (**AOP**, por sus siglas en inglés: **Aspect-Oriented Programming**) es un paradigma de programación que tiene como objetivo modularizar aspectos transversales o preocupaciones que afectan a múltiples partes de un sistema.

Aspectos transversales hace referencia a funcionalidades o preocupaciones que afectan a varias partes del sistema, pero que no están directamente relacionadas con la lógica de negocio central o principal de la aplicación. Estas funcionalidades suelen ser necesarias en múltiples módulos o componentes del sistema, y se "cruzan" o "atraviesan" a través del código.

AOP complementa a la Programación Orientada a Objetos (OOP) al permitir que ciertos comportamientos (como los mencionados) sean definidos en módulos separados, llamados aspectos, sin tener que modificar el código del núcleo de la aplicación. Esto ayuda a reducir la duplicación de código y hace que las aplicaciones sean más modulares y fáciles de mantener.

Un ejemplo, imagina que estás escribiendo un programa con bastante código repetido. Supongamos que estás construyendo un sitio web y debes realizar verificaciones de seguridad en cada página. Podrías escribir el código de seguridad en cada página por separado; sin embargo, eso sería ineficiente y difícil de gestionar. En su lugar, podrías usar AOP para crear un "aspecto de seguridad" separado que maneje las verificaciones de seguridad para todas las páginas en un solo lugar centralizado. Esto hace que el código esté más organizado y sea más fácil de modificar en el futuro.

### 6.2.2 Conceptos principales de AOP

**Aspect:** Un módulo que contiene un comportamiento transversal. Los aspectos definen qué funcionalidad se debe agregar a un sistema sin alterar su estructura original.

**Joinpoint:** Un punto en la ejecución del programa (por ejemplo, la llamada a un método o la ejecución de un bloque de código) donde se puede aplicar un aspecto.

**Advice:** La acción que se toma en un joinpoint. El advice puede ejecutarse antes, después o en lugar de la ejecución del método original, según el tipo de advice que se utilice.

**Pointcut:** Una expresión que define un conjunto de joinpoints en los cuales un aspecto debe ser aplicado. Permite seleccionar de manera flexible y precisa los puntos del programa donde se insertarán los aspectos.

**Weaving:** El proceso de aplicar los aspectos al código fuente o a los archivos binarios. Este proceso puede ocurrir en tiempo de compilación, carga de clase o ejecución.

### **6.2.3 Introducción a la tarea**

Esta tarea consistió en aplicar el paradigma de Programación Orientada a Aspectos (AOP) para gestionar ciertos mensajes de logging dentro del servicio Message Extractor. Para ello, se utilizó el framework AspectJ, una de las implementaciones más conocidas de AOP en Java.

**[INFORMACIÓN CONFIDENCIAL]**

**[FIN INFORMACIÓN CONFIDENCIAL]**

### **6.2.4 Conclusión de la tarea**

El uso de *Aspect-Oriented Programming* (AOP) en este caso nos proporciona una solución elegante para tareas repetitivas sin ensuciar la lógica de negocio del código principal. De esta forma, cualquier cambio futuro en la implementación del procesamiento de los mensajes se puede realizar de manera centralizada en el *Aspect*, sin necesidad de modificar el código principal de la aplicación. Esto mejora la modularidad, legibilidad y flexibilidad del sistema, facilitando su mantenimiento y extensión en el futuro.

**6.3 Investigate CAM compatibility between camV2 and camRel21**

**Fechas de la tarea:** 10/03/2025-14/03/2025

**[INFORMACIÓN CONFIDENCIAL]**

**[FIN INFORMACIÓN CONFIDENCIAL]**

#### **6.4 CDD-V4 Implementation**

**Fechas de la tarea:** 17/03/2025-10/04/2025

**[INFORMACIÓN CONFIDENCIAL]**

**[FIN INFORMACIÓN CONFIDENCIAL]**

**6.5 Get ASN1Type from DF in adaptationLib new CAM version**

**Fechas de la tarea:** 23/04/2025-14/05/2025

**[INFORMACIÓN CONFIDENCIAL]**

**[FIN INFORMACIÓN CONFIDENCIAL]**

## 7 Conclusión

La realización de este Trabajo de Fin de Grado ha sido una experiencia positiva. Por un lado, el estudio y desarrollo de plataformas ITS y, más concretamente, de sistemas cooperativos C-ITS, me ha permitido adentrarme en un campo tecnológico avanzado y en constante evolución, donde la comunicación y la automoción convergen para ofrecer soluciones inteligentes al transporte del futuro. Nunca habría imaginado, siendo aún estudiante en prácticas, que tendría la oportunidad de participar en el desarrollo de tecnologías innovadoras con potencial de impacto en la movilidad del mañana.

Por otro lado, este proyecto me ha ofrecido una visión práctica del entorno profesional de la programación, donde se aplican un amplio conjunto de metodologías, herramientas y buenas prácticas que favorecen un flujo de trabajo eficiente y colaborativo. Uno de los aspectos más valiosos ha sido precisamente el trabajo en equipo. Muchas veces, un simple consejo o una sugerencia de un compañero ha supuesto la solución rápida y eficaz a un problema que de otro modo habría llevado horas resolver por mi cuenta.

En cuanto a los objetivos propuestos en este trabajo, puedo afirmar con satisfacción que se han cumplido con éxito. He participado activamente en el desarrollo —aunque parcial— de una plataforma ITS que sigue los estándares internacionales y que actualmente está siendo utilizada en las pistas internas de la empresa, algunas de las cuales aún no se han desplegado en escenarios reales. Saber que, en el futuro, algunas de estas aplicaciones formarán parte de los vehículos conectados en circulación y que he contribuido, aunque sea de forma modesta, a su desarrollo, es una fuente de orgullo profesional.

Durante el proceso de aprendizaje y desarrollo de la plataforma ITS de Applus+ IDIADA, he tenido la oportunidad tanto de aprender a usar nuevas herramientas y conocimientos técnicos, como de reforzar y afianzar aquellos que ya había aprendido durante la carrera.

- **Java**, como lenguaje principal de programación, reforzando los conocimientos adquiridos durante el grado.
- **Spring Boot**, entendiendo lo básico.
- **Maven**, herramienta de gestión de dependencias y automatización de construcción de proyectos.
- **WSL2**, entorno que permite ejecutar contenedores Linux en Windows y desplegar múltiples servicios localmente.
- **Docker y Kubernetes**, para la orquestación y gestión visual de clústeres de microservicios.
- **MQTT**, protocolo de mensajería ligero usado en la plataforma para gestionar la distribución de mensajes mediante brokers y suscripciones temáticas.
- **JUnit**, empleado para la creación de tests unitarios y tests de integración. La importancia de diseñar tests exhaustivos y de conocer en detalle las especificaciones de los protocolos que usamos para evitar arrastrar errores.
- **SonarQube**, utilizado para evaluar la cobertura de tests, detectar malas prácticas y debilidades en el código.
- **Jenkins**, como herramienta de integración y entrega continua (CI/CD), automatizando el proceso de compilación, testeo y despliegue del software.

**Palabras clave**

1. ITS: Intelligent Transportation Systems
2. C-ITS: Cooperative Intelligent Transportation Systems
3. V2X: Vehicle-to-Everything
4. C-V2X: Cellular Vehicle-to-Everything
5. CAM: Cooperative Awareness Message
6. DENM: Decentralized Environmental Notification Message
7. IVIM: In-Vehicle Information Message
8. MAPEM: MAP Extended Message
9. SPATEM: Signal Phase and Timing Extended Message
10. CPM: Collective Perception Message
11. MCM: Maneuver Coordination Message
12. VRU: Vulnerable Road Users
13. VAM: VRU Awareness Message
14. SREM: Signal Request Extended Message
15. SSEM: Signal Status Extended Message
16. REST: Representational State Transfer
17. API: Application Programming Interface
18. DTO: Data transfer object
19. MQTT: Message Queuing Telemetry Transport
20. IoT: Internet of Things
21. V2V: Vehicle-to-Vehicle
22. GPS: Global Positioning System
23. ADAS: Advanced Driver Assistance Systems
24. V2P: Vehicle-to-Pedestrian
25. V2I: Vehicle-to-Infrastructure
26. SAE: Society of Automotive Engineers
27. DSRC: Dynamic Short-Range Communications
28. WAVE: Wireless Access in Vehicular Environments
29. OSI: Open Systems Interconnection
30. ETSI: European Telecommunications Standards Institute
31. 3GPP: 3rd Generation Partnership Project
32. V2N: Vehicle-to-Network
33. LTE: Long Term Evolution
34. DF: Data Frame
35. DE: Data Element
36. ASN.1: Abstract Syntax Notation One
37. UPER: Unaligned Packed Encoding Rules
38. CDD: Common Data Dictionary
39. JWT: JSON Web Token
40. WSL: Windows Subsystem for Linux
41. AOP: Aspect-Oriented Programming
42. AWS: Amazon Web Services

## Referencias

- [1] Página web: <https://es.mtc.ca.gov/operations/programs-projects/intelligent-transportation-systems-its>
- [2] Página web: <https://www.aleatica.com/tecnologias-sistemas-de-transporte-inteligentes/>
- [3] Página web: <https://www.transportes.gob.es/transporte-terrestre/sistemas-inteligentes-de-transporte/sistemas-inteligentes-de-transporte-its>
- [4] Página web: <https://www.wsp.com/es-mx/insights/los-sistemas-de-transporte-inteligentes-eficaces-integran-factores-humanos>
- [5] Página web: <https://es.digi.com/blog/post/introduction-to-smart-transportation-benefits>
- [6] Artículo de revista: Suárez Flórez, M., “Los sistemas inteligentes de transporte ITS”, Ciencia e Ingeniería Neogranadina, 2001, n.º 10, pp. 39–45
- [7] Artículo de revista: Wilbur, M.; Sivagnanam, A.; Ayman, A.; Samaranayake, S.; Dubey, A.; Laszka, A., “Artificial Intelligence for Smart Transportation”, arXiv preprint arXiv:2308.07457, 2023
- [8] Artículo de revista: Zuluaga-Isaza, J. A.; Arango-Monsalve, A. M., “Una revisión a los sistemas de transporte inteligente”, Revista de Ingenierías Interfaces, 2023, Vol. 6, n.º 1, pp. 1–18
- [9] Página web: <https://www.dgt.es/muevete-con-seguridad/tecnologia-e-innovacion-en-carretera/sistemas-inteligentes-de-transporte-its/>
- [10] Página web: <https://revista.une.org/48/las-normas-impulsan-los-sistemas-inteligentes-de-transporte.html>
- [11] Página web: <https://codegym.cc/es/groups/posts/es.543.que-es-aop-principios-de-la-programacion-orientada-a-aspectos>
- [12] Página web: <https://www.spiceworks.com/tech/devops/articles/what-is-aop/>
- [13] Página web: [https://www.etsi.org/deliver/etsi\\_en/302600\\_302699/30263603/01.01.02\\_20/en\\_30263603v010102a.pdf](https://www.etsi.org/deliver/etsi_en/302600_302699/30263603/01.01.02_20/en_30263603v010102a.pdf)
- [14] Página web: <https://www.kimley-horn.com/news-insights/perspectives/dsrc-cv2x-comparison-future-connected-vehicles/>
- [15] Página web: <https://www.nxp.jp/docs/en/white-paper/CITSCOMPWP.pdf#:~:text=1,14>
- [16] Página web: <https://www.arc-it.net/html/standards/standard469.html#:~:text=This%20document%20defines%20the%20OSI,range%205%2C905%20GHz%20to%205%2C925>
- [17] Página web: <https://www.car-2-car.org/about-c-its#:~:text=cooperative%20V2X%20communication%20uses%20the,at%20any%20time%2C%20data%20transmission>
- [18] Página web: <https://www.gsma.com/solutions-and-impact/technologies/internet-of-things/wp-content/uploads/2017/09/GSMA-position-on-CV2X.pdf#:~:text=Wrong%20way%20driving%20warning%20Traffic,This%20model%20can%20be>
- [19] Página web: <https://5gaa.org/content/uploads/2018/10/Position-Paper-ITG5.pdf#:~:text=mobile%20communication%20networks%3B%20b,supported%20by%20spectrum%20that%20is>
- [20] Página web: <https://es.digi.com/resources/definitions/mqtt#:~:text=MQTT%20explicado%3A%20MQTT%20funciona%20seg%3BAN.es%20necesario%20minimizar%20la%20latencia>
- [21] Página web: <https://www.paessler.com/es/it-explained/mqtt#:~:text=MQTT%20se%20ejecuta%20sobre%20TCP%20FIP,editor%2C%20un%20suscriptor%20o%20ambos>
- [22] Página web:

- <https://docs.aws.amazon.com/whitepapers/latest/designing-next-generation-vehicle-communication-aws-iot/mqtt-for-connected-vehicle-platforms.html#:~:text=Unlike%20a%20traditional%20client,subscribed%20to%20a%20specific%20topic>
- [23] Página web: <https://www.embedded.com/applying-mqtt-for-the-internet-of-vehicles/#:~:text=In%20this%20model%2C%20publishers%20send,%E2%80%9Cwildcards%2C%E2%80%9D%20it%20consolidates%20subscriptions%20accordingly>
- [24] Página web: <https://www.emqx.com/en/blog/mqtt-for-internet-of-vehicles>
- [25] Página web: <https://reflectoring.io/spring-hexagonal/>
- [26] Página web: <https://medium.com/ssense-tech/hexagonal-architecture-there-are-always-two-sides-to-every-story-bc0780ed7d9c>
- [27] Página web: <https://www.itsstandards.eu/app/uploads/sites/14/2020/10/C-ITS-Brochure-2020-FINAL.pdf#:~:text=,limits%2C%20car%20sharing%3B%20%E2%80%94%20comfort>
- [28] Página web: [etsi.org/deliver/etsi\\_ts/103900\\_103999/103900/02.02.01\\_60/ts\\_103900v020201p.pdf](https://etsi.org/deliver/etsi_ts/103900_103999/103900/02.02.01_60/ts_103900v020201p.pdf)
- [29] Página web: <https://grupobim.com/en/blog/ventajas-arquitectura-hexagonal-java/#:~:text=1.de%20frameworks%20o%20infraestructuras%20externas>
- [30] Página web: <https://www.baeldung.com/hexagonal-architecture-ddd-spring>
- [31] Página web: [https://www.etsi.org/deliver/etsi\\_en/302600\\_302699/30263603/01.01.02\\_20/en\\_30263603v010102a.pdf](https://www.etsi.org/deliver/etsi_en/302600_302699/30263603/01.01.02_20/en_30263603v010102a.pdf)
- [32] Página web: [https://www.etsi.org/deliver/etsi\\_ts/102800\\_102899/10289402/02.03.01\\_60/ts\\_10289402v020301p.pdf](https://www.etsi.org/deliver/etsi_ts/102800_102899/10289402/02.03.01_60/ts_10289402v020301p.pdf)