Grant Agreement Number: 731993

Project acronym: AUTOPILOT

Project full title: AUTOmated driving Progressed by Internet Of Things

D.2.4
Report on development and Integration of IoT devices into IoT ecosystem

Due delivery date: 30 June 2018

Actual delivery date: 03 July 2018

Organization name of lead participant for this deliverable: SINTEF

Project co-funded by the European Commission within Horizon 2020 and managed by the European GNSS Agency (GSA)		
Dissemination level		
PU	Public	X
PP	Restricted to other programme participants (including the GSA)	
RE	Restricted to a group specified by the consortium (including the GSA)	
CO	Confidential, only for members of the consortium (including the GSA)	

Project funded by the European Union's Horizon 2020 Research and Innovation Programme (2014 – 2020)

AUTOPILOT

Document Control Sheet

Deliverable number:	2.4
Deliverable responsible:	SINTEF
Work package:	2
Editor:	Ovidiu Vermesan

Author(s)		
Name	**Organisation**	**E-mail**
Ovidiu Vermesan	SINTEF	ovidiu.vermesan@sintef.no
Roy Bahr	SINTEF	roy.bahr@sintef.no
Marcus Mueller	DLR	Marcus.Mueller@dlr.de
Philipp Lutz	DLR	Philipp.lutz@dlr.de
Louis Touko Tcheumadjeu	DLR	Louis.toukotcheumadjeu@dlr.de
Mariano Falcitelli	CNIT	mariano.falcitelli@cnit.it
Paolo Pagano	CNIT	paolo.pagano@cnit.it
Sandro Noto	CNIT	sandro.noto@cnit.it
Michel Yeung	CONTI	Michel.Yeung@continental-corporation.com
Jean-Francois Simeon	CONTI	jean-francois.simeon@continental-corporation.com
Yassine Banouar	CONTI	yassine.banouar@continental-corporation.com
Jose Manuel Martinez	CTAG	josemanuel.martinez@ctag.com
Xurxo Legaspi Ramos	CTAG	xurxo.legaspi@ctag.com
Silvia Alén González	CTAG	silvia.alen@ctag.com
Anton Dekusar	IBMIE	ADekusar@ie.ibm.com
Alexander Velizhev	IBMRE	AVE@zurich.ibm.com
Enrico Ferrera	ISMB	ferrera@ismb.it
Ilaria Bosi	ISMB	bosi@ismb.it
Daniele Brevi	ISMB	brevi@ismb.it
Gurkan Solmaz	NEC	gurkan.solmaz@neclab.eu
Mahdi Ben Alaya	SEN	benalaya@sensinov.com
Vincenzo Di Massa	THA	vincenzo.dimassa@thalesgroup.com
Carlotta Firmani	THA	carlotta.firmani@thalesgroup.com
Jos den Ouden	TUE	j.h.v.d.ouden@tue.nl
Johan Scholliers	VTT	johan.scholliers@vtt.fi
Floriane Schreiner	VED	floriane.schreiner@vedecom.fr
Bram van den Ende	TNO	bram.vandenende@tno.nl
Stella Nikolaou	CERTH	snikol@certh.gr
Herve Marcasuzaa	VALEO	herve.marcasuzaa@valeo.com

Document Revision History			
Version	**Date**	**Modifications Introduced**	
		Modification Reason	**Modified by**
V0.01	30/06/2017	Template integration and initial version.	SINTEF
V0.02	10/10/2017	Activities description and initial partners contributions.	SINTEF
V0.03	27/11/2017-28/06/2018	The sections 2, 3, 4, 5, 6, 7, 8	SINTEF

V0.04	01/12/2017-27/06/2018	The sections 2, 3, 4, 5, 6, and 7 (Spanish pilot site).	CTAG
V0.05	01/12/2017-27/06/2018	The sections 2, 3, 4, 5, 6, 7, and 8 (Dutch pilot site).	TNO, DLR, TUE, NEC, IBMIE, IBMRE, VALEO
V0.06	12/12/2017-27/06/2018	The sections 2, 3, 4, 5, 6, 7 and 8 (Italian pilot site).	CNIT, ISMB, THA
V0.07	15/12/2017-26/06/2018	The sections 2, 3, 4, 5, and 7 (French pilot site).	CONTI, SEN, VED, CERTH
V0.08	01/03/2018-26/06/2018	The sections 2, 3, 4, 5, 6, and 7 (Finnish pilot site).	VTT
V0.09	15/06/2018	Integrated the received inputs from the pilot sites partners. Send for the first internal review.	SINTEF
V0.10	22/06/2018	Integrated the received inputs from the pilot sites partners. Send for the second internal review.	SINTEF
V0.11	27/06/2018	The sections 1 and 9 (Executive summary and Discussions)	SINTEF
V0.12	27/06/2018	Updates according to internal reviews and final T2.4 telco.	SINTEF
V0.13	28/06/2018	Update sections 7 and 8.	SINTEF, DLR, TNO
V0.14	28/06/2018	Final check.	SINTEF, DLR
V1.00	29/06/2018	Release.	ERTICO

Abstract

This report describes the development and integration of IoT devices contributing to autonomous driving – both new and existing devices adapted to become IoT devices. Mobile IoT objects (mobile robots and/or micro aerial vehicles) and IoT infrastructure (sensor/actuators, connectivity and communication) developed and seamlessly integrated into the IoT ecosystem (other IoT devices, vehicle IoT platform developed in T2.1 and Open IoT platform developed in T2.3) are presented and the details of the use cases and functions for IoT devices, the interfaces, communication, security and platforms integration are highlighted.

Legal Disclaimer

Abbreviations and Acronyms

Acronym	Definition
6LoWPAN	IPv6 over Low-Power Wireless Personal Area Networks
ACC	Active cruise control
AD	Autonomous driving
AEB	Automatic emergency braking
AP	Automatic pilot
API	Application Programming Interface
App	Application
ASN	Application Service Node
AVP	Automated Valet Parking
BLE	Bluetooth Low Energy
CAD	Connected and Automated Driving
CAM	Cooperative Awareness Message
CC	Control Centre
CeH	Connected e-Horizon
C-ITS	Cooperative Intelligent Transportation Systems
CoAP	Constrained Application Protocol
CSE	Common Service Node
DB	Database
DDS	Data Distribution Service
DENM	Decentralized Environmental Notification Message
DM	Driver monitoring
DSRC	Dedicated Short-Range Communications
EC	European Commission
FR	Functional requirements
GA	Grant Agreement
GPS	Global positioning system
GPU	Graphics processing unit
GW	Gateway
HMI	Human Machine Interface
HTTP	Hypertext transfer protocol
IMU	Inertial measurement unit
IoT	Internet of things
IoV	Internet-of-Vehicles
ITS	Intelligent Transport Systems
LDWS	Lane departure warning system
LKA	Lane keep assist
LRR	Long-range radar
LTE	Long-Term Evolution
M2M	Machine-to-Machine
MAP	Map Data
MAV	Micro aerial vehicle
MQTT	Message Queuing Telemetry Transport
MRR	Medium-range radar
NB-IoT	Narrowband IoT
NFR	Non-functional requirements
OBU	On-Board Unit

OCB mode	Offset Codebook Mode
OEM	Original Equipment Manufacturer
OSGi	Open Services Gateway initiative
PA	Park assist
PDF	probability density function
PF	Platform
PMS	Parking Management Service
PO	Project officer
PoI	Point of Interest
PS	Pilot site
RSU	Road Side Unit
SISCOGA	SIStemas COoperativos de GAlicia
SDK	Software development kit
SPaT	Single Phase and Time
SRR	Short-range radar
TCC	Traffic Control Centre
ToF	Time of Flight
UCR	Use case requirements
UWB	Ultra-Wide Band
V2D	Vehicle-to-Device
V2G	Vehicle-to-Grid
V2I	Vehicle-to-Infrastructure
V2P	Vehicle-to-Pedestrian
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VRU	Vulnerable Road User
VVT	Verification, validation and testing
WP	Work Package
w.r.t.	With regard to

AUTOPILOT

Table of Contents

Executive Summary **14**

1. Introduction **16**

1.1 Purpose of the document 16

1.2 Intended audience 16

2. Autonomous vehicles domains **17**

2.1 Vehicle to vehicle (V2V) domain 18

1.1.1 French pilot site 18

1.1.2 Dutch pilot site 18

1.1.3 Italian pilot site 19

1.1.4 Spanish pilot site 19

1.1.5 Finnish pilot site 19

2.2 Vehicle to pedestrian (V2P) domain 20

1.1.6 French pilot site 20

1.1.7 Dutch pilot site 21

1.1.8 Italian pilot site 21

1.1.9 Spanish pilot site 21

1.1.10 Finnish pilot site 21

2.3 Vehicle to device (V2D) domain 21

1.1.11 French pilot site 21

1.1.12 Dutch pilot site 22

1.1.13 Italian pilot site 22

1.1.14 Spanish pilot site 23

1.1.15 Finnish pilot site 23

2.4 Vehicle to grid (V2G) domain 23

2.5 Vehicle to infrastructure (V2I) domain 23

1.1.16 French pilot site 23

1.1.17 Dutch pilot site 24

1.1.18 Italian pilot site 26

1.1.19 Spanish pilot site 26

1.1.20 Finnish pilot site 27

3. IoT sensors and actuators for autonomous vehicle applications **29**

3.1 Ultrasonic 31

1.1.21 Spanish pilot site 31

3.2 Radar 31

1.1.22 Long-range radar (LRR) 31

1.1.23 Medium-range radar (MRR) 31

1.1.24 Spanish pilot site 31
1.1.25 Short-range radar (SRR) 31
3.3 Optical 32
1.1.26 Long distance camera 32
1.1.27 Spanish pilot site 32
1.1.28 Smart Camera 32
1.1.29 Spanish pilot site 32
1.1.30 Finnish pilot site 33
1.1.31 Stereo camera 33
1.1.32 Italian pilot site 33
1.1.33 Dutch pilot site 36
3.4 Lidar 39
1.1.34 Spanish pilot site 39
1.1.35 Dutch pilot site 39
1.1.36 Italian pilot site 40
1.1.37 Finnish pilot site 41
3.5 Accelerometer 41
1.1.38 Dutch pilot site 42
1.1.39 Italian pilot site 42
1.1.40 Smartphone Accelerometer 42
1.1.41 6LoWPAN Inertial Sensor 44
1.1.42 Inertial Measurement Unit accelerometer sensor 45
3.6 UWB localization 46
1.1.43 Dutch pilot site 46
1.1.44 Finnish pilot site 46
3.7 Crowd detector device with Wi-Fi and GPS sensors 46
1.1.45 Dutch pilot site 46
3.8 BLE Beacons 48
1.1.46 French pilot site 48
1.1.47 Dutch pilot site 48
3.9 NB-IoT sensors 49
1.1.48 Italian pilot site 49
3.10 LoRaWAN 52
1.1.49 French pilot site 52
4. Use cases and functions for IoT devices and the pilot sites 53
4.1 French pilot site 54
1.1.50 Car sharing use case functions (Versailles) 54
1.1.51 Platooning use case functions (Versailles) 55

1.1.52 Urban driving use case functions (Versailles) 55
4.2 Dutch pilot site 56
1.1.53 Automated valet parking use case functions (Brainport) 56
1.1.54 Highway pilot use case functions (Brainport) 57
1.1.55 Platooning use case functions (Brainport) 58
1.1.56 Urban driving / Car rebalancing use case functions (Brainport) 60
1.1.57 Car sharing use case functions (Brainport) 60
4.3 Italian pilot site 61
1.1.58 Highway pilot use case functions (Livorno-Florence) 62
1.1.59 Urban driving use case functions (Livorno-Florence) 66
4.4 Spanish pilot site 67
1.1.60 Automated valet parking use case functions (Vigo) 68
1.1.61 Urban driving use case functions (Vigo) 69
4.5 Finnish pilot site 70
1.1.62 Automated valet parking use case functions (Tampere) 71
1.1.63 Urban driving use case functions (Tampere) 71
5. IoT platforms and IoT devices integration 72
5.1 IoT platform and IoT devices integration - Versailles 72
5.2 IoT platform and IoT devices integration - Brainport 73
5.3 IoT platform and IoT devices integration - Livorno-Florence 75
5.4 IoT platform and IoT devices integration - Vigo 75
5.5 IoT platform and IoT devices integration - Tampere 77
6. Communication, security and platform interfaces 78
6.1 Dutch pilot site 78
1.1.64 Communication interfaces 80
6.2 Italian pilot site 80
1.1.65 Integration and communication between 6LoWPAN devices, CNIT IoT-G5 and OneM2M platform 87
1.1.66 Security features for IoT devices and IoT communication interfaces 91
6.3 Spanish pilot site 92
1.1.67 oneM2M communication interface 92
1.1.68 IBM Watson IoT devices communication platform 92
6.4 Finnish pilot site 93
7. Application integration 94
7.1 Application Integration - Versailles 94
7.2 Application integration - Brainport 95
1.1.69 Automated valet parking use case 95
1.1.70 Car sharing use case 98

1.1.71 Urban driving / Car rebalancing use case 99
1.1.72 Platooning use case 100
1.1.73 Highway Pilot use case 101
7.3 Application integration – Livorno-Florence 102
7.4 Application integration – Vigo 104
1.1.74 Urban driving use case 104
1.1.75 Automated valet parking use case 104
7.5 Application integration – Tampere 105
8. Verification, validation and testing for selected cases 107
8.1 Integration of devices/applications to oneM2M IoT Platform 107
8.2 Smartphone connection and testing with OneM2M and HUAWEI OceanConnect 108
8.3 NB-IoT device 109
9. Conclusions and discussions 112
10. References 114

List of Figures

Figure 1 – Automated vehicle domains 17
Figure 2 - Additional vehicle HW/SW in the Prius vehicle used for Platooning and AVP (Dutch Pilot site) 19
Figure 3 – Architecture of the V2D bridge connecting to a WSN 22
Figure 4 - Traffic light assist architecture for platooning in Versailles (complicated crossroads) 23
Figure 5 - System view of Highway Pilot in Brainport 24
Figure 6 - Interaction between AVP devices and IoT platforms 25
Figure 7 – Automated vehicle ecosystem 29
Figure 8 – Sensors, actuators and functions 30
Figure 9 – Radar observation ranges [5] 31
Figure 10 – Smart camera (UI-1221-LE, IDS GmbH) 33
Figure 11 - HIKVISION DS-2DF8223I-AEL camera used in the Finnish pilot site 33
Figure 12 – ZED camera from Stereolabs 35
Figure 13 – Nvidia Jetson TX2 Development Board 36
Figure 14 – MAV camera setup. Blue illustrates the field of view of the stereo cameras 37
Figure 15 - AVP road unit camera and parking lot detection visualization 39
Figure 16 – DLR prototype, position of laser scanners 39
Figure 17 - Position of LiDAR on the TNO vehicle 40
Figure 18 – ITRI prototype, VRU Protection with LIDAR 41
Figure 19 – Gravity vector and heading, pitch and roll about axes 42
Figure 20 – Downward jerk sensed by accelerometer which occurred due to potholes over the road. 42
Figure 21 – Coordinate system (relative to a device) that's used by the Sensor API. 43
Figure 22 – Wireless sniffer receiving wireless signals from VRU. 47
Figure 23 – An overview of the NEC crowd detector device. 47
Figure 24 - BLE beacon used for the Point of Interest notifications 48
Figure 25 – Beacon devices: nRF51822 and Raspberry Pi Zero W. 48
Figure 26 – NB-IoT scenario 49
Figure 27 – NB-IoT channels 50
Figure 28 – Water level sensor model connected to the NB-IoT puddle detector 50
Figure 29 – Processing of the measured level of water 50
Figure 30 – NB-IoT smart object developed by CNIT. 1) Quectel BC95-B20 transceiver; 2) Evaluation board; 3) Management board; 4) Battery; 5) Power input; 6) On/Off button; 7) Sensor input; 8) Antenna 51
Figure 31 - Parking detector (ONESITU) used on car sharing stations 52
Figure 32 - Versailles pilot site architecture overview 54
Figure 33 - The Brainport pilot architecture (simplified) 56
Figure 34 - Automated valet parking use case IoT architecture 57
Figure 35 - Software and IoT architecture in Valeo prototypes 58
Figure 36 - Platooning use case execution view of Platoon formation 59
Figure 37 - GUI of the Traffic Manager application (TASS) 60
Figure 38 – Car Sharing Use Case Architecture 61
Figure 39 – Livorno pilot site architecture 62
Figure 40 – Hazard on the roadway (puddle) execution view 64
Figure 41 – Roadworks warning by TCC execution view 65
Figure 42 – Urban driving execution view 67
Figure 43 – Vigo pilot site architecture 67
Figure 44 – Valet parking execution view 69
Figure 45 – Urban AUTOPILOT execution view 70

Figure 46 - Architecture of the Finnish pilot 71
Figure 47 – IoT platform and IoT Versailles integration 72
Figure 48 – Target platform integration 73
Figure 49 – Wireless sniffer integration with FIWARE IoT Platform. 74
Figure 50 – Integration of BLE beacon to the FIWARE-based IoT Platform. 74
Figure 51 – Integration of IoT devices into the oneM2M IoT platform at the Italian PS 75
Figure 52 – Vigo IoT platform and IoT devices integration 76
Figure 53 – Architecture of the Finnish pilot 77
Figure 54: Technical architecture Brainport, emphasizing the physical connections 78
Figure 55 - Platform interfaces in Dutch Pilot site Brainport 79
Figure 56 – In vehicle software architecture for IoT platform integration (Italian pilot site) . 81
Figure 57 – ISMB IoT in vehicle platform 81
Figure 58 – Italian PS smart traffic light 83
Figure 59 – Road side unit (RSU) software architecture for IoT platform integration (Italian pilot site) 84
Figure 60 – In vehicle platform, runtime environment (Italian pilot site) 84
Figure 61 – Road side unit (RSU) IoT platform, runtime environment (Italian pilot site) 85
Figure 62 – NB-IoT/OneM2M platform communication scheme 86
Figure 63 – CNIT IoT-G5 87
Figure 64 – 6LowPAN/IoT-G5/OneM2M platform communication and integration scheme . 88
Figure 65 – Vehicles/IoT-G5/OneM2M platform communication and integration scheme. .. 91
Figure 66: Spanish pilot site communication architecture. 92
Figure 67. Finnish pilot communication architecture 93
Figure 68 – IoT services for Versailles pilot site 94
Figure 69 – High-level features of IoT services 95
Figure 70 - AVP IoT Information flow and usage Part I 96
Figure 71 - AVP IoT Information flow and usage (Part II) 97
Figure 72 – Architecture diagram for the car sharing use case 99
Figure 73 – IoT device and IoT platform integration diagram for Brainport Urban Driving / Car Rebalancing use case 100
Figure 74 - IoT device and IoT platform integration diagram for Brainport Platooning use case 101
Figure 75 - Hazard Information in Google Maps 102
Figure 76 – Livorno Pilot Site Overall Architecture 103
Figure 77 – Architecture diagram for the parking service use case. 105
Figure 78 – Mechanisms of connecting oneM2M and Non-oneM2M devices/applications into the IoT platform 108
Figure 79 – Required IPE per pilot site overview 108
Figure 80 – The SmartCampus structure on OneM2M 109
Figure 81 – Communication scheme between NB-IoT device and ICON oneM2M platform. 110
Figure 82 – TIM ICON oneM2M platform resource tree. 110

List of Tables

Table 1 - Overview about the IoT sensors used in different pilot sites 30
Table 2 – Smart camera characteristics 32
Table 3 – ZED Stereo Camera Specifications 35
Table 4 – Nvidia Jetson TX2 Development Board Specifications 36
Table 5 – Lidar installed in the Jeep Renegade 40
Table 6 – SensOne inertial sensor characteristics 44
Table 7 - Integrated electronic compass 45
Table 8 – Water detector main features 50

Table 9 – Quectel BC95-B20 key features 51
Table 10 – Pilot sites and use cases 53
Table 11 - SAE automation levels [25] 53
Table 12 - Overview relevant interfaces 78
Table 13 - Overview relevant interfaces 79
Table 14 – NB-IoT smart object Data Model 86
Table 15 – RSU Data Model for LoRa/6LowPAN network sensors 89
Table 16 - Overview about the AVP IoT data used in the Brainport pilot site 97

Executive Summary

Connectivity is part of the major automotive innovation for the foreseeable future and the connected vehicles integrate communication technologies to eliminate potentially fatal distractions. It enables cars to communicate with other vehicles, infrastructure, pedestrians, electric grids and devices, to optimize the driving environment. Various connectivity protocols play a key role in this vision and, as the level of connectivity increases, the IoT applications are becoming more and more important.

IoT devices support autonomous driving functions – such as notification about a parking spot being made available – which require that the IoT device connectivity covers wider areas and require less latency.

According to the AUTOPILOT partners the IoT devices and technologies can support the autonomous driving functions in different ways. They show that the development and integration of IoT devices into combined autonomous vehicles and IoT ecosystems could support the integration of services using interoperable IoT platforms and IoT devices that provide additional information to the vehicles about the environment, surroundings and the dynamic events around the vehicles.

This document describes the development of IoT devices and infrastructure and their seamless integration into the IoT ecosystem for automated driving and increased functionality.

Mobile IoT objects (e.g. micro aerial vehicle (MAV)) and IoT infrastructure (sensors/actuators, connectivity and communication) developed in the project and integrated into the IoT ecosystems deployed in different pilot sites (other IoT devices, vehicle IoT platform developed in T2.1 and Open IoT platform developed in T2.3) are defined and presented.

The overall activities are based on the requirements and specifications presented in the Use Case specification (T1.1), IoT architecture (T1.2), IoT vehicle specifications (T1.3) and communication specifications (T1.4).

Chapter 2 briefly presents the five domains and describes what type of AUTOILOT use cases address the different domains, and the type of IoT devices used in the respective domains.

Chapter 3 outlines the IoT devices, sensors and actuators used and/or developed by AUTOPILOT partners in the different use cases and integrated into the IoT ecosystem for the autonomous vehicle applications functions.

Chapter 4 describes the functions of IoT devices and the use cases in which they are implemented in the five pilot sites (Brainport, Livorno, Tampere, Versailles, and Vigo). The Korean pilot site in Daejeon is not described in deliverable D2.4 because no IoT platform has been implemented (see also deliverable D2.3). The description of the IoT device functions and the use cases includes connectivity between IoT devices, connectivity between vehicles, infrastructure and other sensors to enhance autonomous driving capabilities and technology that allows vehicles to monitor the state and availability of different services. As for the in-vehicle functions, three main groups of sensor systems such as camera-, radar-, and lidar-based systems, together with ultrasonic sensors are used for autonomous driving, Chapter 4 presents how other type of sensors/actuators and IoT devices in the different use cases are used to enhance the autonomous driving capabilities.

Chapter 5 presents the integration of IoT devices into IoT end-to-end platforms that provide the hardware, software, connectivity, security, and device management tools to handle the different IoT devices used in the different use cases across the AUTOPILOT pilot sites. Different sections provide info on how the integration is implemented presenting the managed integrations, device management, cloud connection, cellular modem, etc. to manage and monitor the IoT devices in different use cases.

Chapter 6 reports on the communication and connectivity implemented in the different use cases

using the IoT devices developed for providing different functions. The description addresses as well the IoT platforms interfaces used and the security mechanisms implemented for the different specific cases and pilot sites.

Chapter 7 addresses the high-level integration of the IoT devices and use cases in the different pilot sites. The information is linked to how the data acquired through the IoT devices and platforms are integrated in different applications, back ends and cloud services.

Chapter 8 describes the verification, validation and testing (VV&T) of selected cases for IoT devices used in autonomous vehicles use cases for achieving acceptable levels of safety and assurance for the autonomous vehicle applications. These are VV&T methods used for the IoT devices, sub-systems, communication, and integration into IoT platforms.

1. Introduction

1.1 Purpose of the document

The document represents the Deliverable D2.4 “Report on development and Integration of IoT devices into IoT ecosystem”, first output carried out within Task 2.4 "Development and Integration of IoT devices".

The purpose of deliverable D2.4 is to present and describe the development and integration of IoT devices and infrastructure for automated driving and increased functionality, seamlessly integrated into the IoT ecosystems developed in the AUTOPILOT project.

The work is based on the specifications and requirements developed in Task 1.1 addressing Use Case specifications, Task 1.2 describing the IoT architecture, Task 1.3 on IoT vehicle specifications and Task 1.4 providing the communication specifications.

1.2 Intended audience

The intended audience for this deliverable is considered to be all the AUTOPILOT participants and in particular, the AUTOPILOT participants involved in WP1 and the partners involved in Task 2.3 “Development of the open IoT service platform” and in Task 2.5 “Pilot Readiness verification”.

The development and integration of IoT devices and infrastructure for automated driving is used in the other tasks of WP2, WP3 and WP4 to support the developments and deployments of IoT technologies in different pilot sites.

2. Autonomous vehicles domains

The concept of Internet of Vehicles (IoV) or Vehicle-to-Everything (V2X) communications applied for autonomous transportation and mobility applications, requires creating mobile ecosystems based on trust, security and convenience to connectivity services and transportation applications in order to ensure security, mobility and convenience to consumer-centric transactions and services [8][11].

A vehicle with automated features must have established interactions with different domains that are interlinked with operational design domain for which a vehicle could have one or multiple systems, one for each operational design domain (e.g. freeway driving, valet parking, urban driving, etc.). Chapter 2 briefly presents the five domains and describes what type of use cases address the different domains and the type of IoT devices used in the respective domains.

In this context for autonomous vehicle applications, five domains are defined as presented in Figure 1. The domains cover the communications of vehicle to everything (V2X) covering vehicle to infrastructure (V2I), vehicle to pedestrian (V2P), vehicle to device (V2D) vehicle to grid (V2G) and vehicle to vehicle (V2V) as important communication building blocks of the IoT ecosystems [8][11]. The domains are described in deliverable D1.7 (Initial specification of Communication System for IoT-enhanced AD).

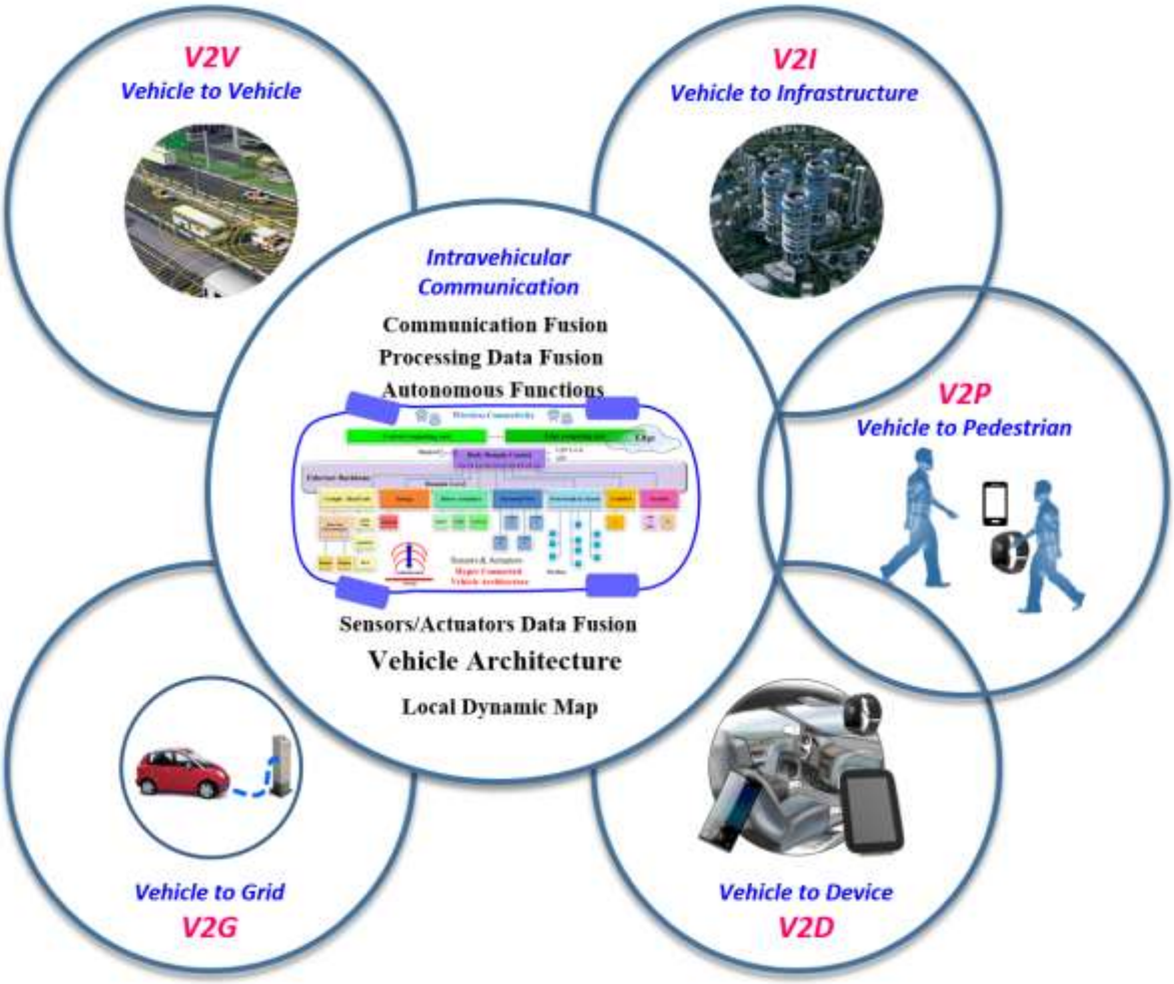

Figure 1 – Automated vehicle domains

Smart sensors and actuators in the vehicles, roads and traffic control infrastructures collect a variety of information to serve enhanced automated driving. These require robust sensor, actuators and communication solutions, which are able to communicate with the control systems while considering the timing, safety and security constraints. Redundancy and parallel systems are required in all safety and security critical applications. It is worth noting that power saving mode (e.g. sensors, actuators) can be a barrier to real-time information. For battery-powered equipment, it will always be a trade-off between power consumption and communication latency [8][9].

The IoT ecosystem relies on interaction among vehicles, pedestrians, devices, micro aerial vehicles (MAVs) and infrastructure, to improve traffic management (increase efficiency, security and safety).

The following chapters present the development and integration of IoT devices contributing to autonomous driving. The mobile IoT objects (mobile robots and/or MAVs) and IoT infrastructure (sensor/actuators, connectivity and communication) are developed and seamlessly integrated into the IoT ecosystem.

Below we describe how the developed/used IoT devices are applied in the different domains for the use cases of each pilot site.

2.1 Vehicle to vehicle (V2V) domain

1.1.1 French pilot site

In the French pilot site, the Vehicle to Vehicle communication is used during the platooning use case. This communication is made over ETSI ITS-G5 (IEEE 802.11 OCB) to allow the exchange of information between the vehicles forming the platoon. This information will be used to place the vehicles according to the others. The goal is to ensure that the platoon is not broken and that all vehicles are capable at any time to cross an intersection for example. All the relevant information about the platoon is displayed to the driver on the embedded screen.

Bicycle OBU: In the French site, the communications between vehicles and bicycles are typical V2V ETSI ITS-G5 communications. The bicycles are equipped with an On-Board-Unit (OBU), which is responsible for ITS-G5 realization. This device has been developed from scratch by CERTH/HIT, specifically for the AUTOPILOT project. The ITS-G5 stack, implementing Cooperative Awareness Messages (CAM) and Decentralised Environmental Messages (DENM) messages, is also developed by CERTH/HIT. Besides V2V, bicycle’s OBU also implements Long-Term Evolution (LTE) communications for IoT information exchange. Cooperative Awareness Message (CAM) and Decentralized Environmental Notification Message (DENM) like messages are sent to the IoT platform via cellular 4G/LTE channels. These messages are then transferred to an external server where the Decision Support System (DSS) and the risk assessment algorithm reside. The outcome of the algorithm is transferred to the bicycles via 4G/LTE channels. Both ITS-G5 and IoT systems will coexist in the bicycle’s OBU.

1.1.2 Dutch pilot site

- Highway: The use case does not implement V2V communication. Information from/to any vehicle is processed by and transmitted through cloud and IoT platforms.
- Platooning: In the Dutch pilot site (PS), Vehicle to Vehicle (V2V) communication is established via ITS-G5 connections (existing technology TNO but adapted to AUTOPILOT) and additionally a UWB connection (novel technology, brought by NXP). An alternative approach for V2V information exchange is via IoT technology, using a local cloud service that forwards messages coming from one vehicle, to another (see also Spanish PS). At the time of writing this was under development at TNO.

Figure 2 - Additional vehicle HW/SW in the Prius vehicle used for Platooning and AVP (Dutch Pilot site)

1.1.3 Italian pilot site

In the Italian PS, Vehicle to Vehicle (V2V) communication is managed by ISMB IoT In-Vehicle platform. The platform implements an almost full ETSI stack, completely developed by ISMB. The main V2V messages that will be exchanged are Cooperative Awareness Messages (CAM). This information will be used to provide a detailed look of the surrounding of the autonomous vehicle. CAM messages notify the information sensed by the In-vehicle sensors to the vehicles in the transmission range. This information is used by the AD data fusion algorithm to take decisions for both the highway and urban scenarios.

For the bicycle to vehicle communications, a prototype was developed by one of the AUTOPILOT partners (ISMB) in 2015. An electric bicycle was equipped with a battery powered On-Board-Unit (OBU) implementing IEEE 802.11p (ETSI ITS-G5) communications. The bicycle OBU exchanges messages (notably CAM) that are used by an anti-collision algorithm to warn the driver and the bicycle rider about a possible forthcoming collision. The rider is warned thanks to a smart-bracelet directly connected with the OBU. The experiment shows good performances also with a basic anti-collision algorithm. A video about this experiment can be found in [14].

1.1.4 Spanish pilot site

In the Spanish Pilot Site, the Vehicle to Vehicle communication is performed using the IoT in-vehicle platform system. In order to check the impact of the fully IoT communication system in the Spanish use cases, this V2V communication is performed through the infrastructure and using standard oneM2M messages. These messages wrap the defined data models that allow the IoT communication with the vehicle. Therefore, if a vehicle needs to send its information to other vehicles, that one will upload the corresponding message to the cloud, which later on will be available for any other connected vehicle.

1.1.5 Finnish pilot site

In the Finnish pilot site, the vehicle is equipped with both ETSI ITS-G5 and 4G/LTE communications. The vehicle ETSI ITS-G5 station transmits CAM messages regularly, and the vehicle position is also sent to the IoT platform, for use by other services.

2.2 Vehicle to pedestrian (V2P) domain

Every year, a large number of Vulnerable Road Users (VRUs) are killed or seriously injured in road accidents. Vehicle communications can significantly reduce the number of these fatalities thanks to an effective warning system for the involved actors.

V2P (Vehicle-to-Pedestrian) communication is a field of research that studies the communications between cars and pedestrians, but it typically considers also bicyclist and motorcyclist, children in strollers, mobility-impaired people with wheelchairs, passengers embarking and disembarking buses, etc. The goal of the V2P is to detect a pedestrian or more generally a VRU, and notify information useful to avoid accidents.

The more intuitive device to warn pedestrians is the smartphone, due to its increasing computational power, the availability of wireless connection and its widespread availability. Today's incumbent standard for vehicular communication is ETSI ITS-G5 that is based on the IEEE 802.11p amendment of Wi-Fi standard. Unfortunately, while Wi-Fi is supported by most smartphones, IEEE 802.11p is not implemented in any commercial product. Since this approach seems interesting, in 2014 Qualcomm and Honda, published a paper [13] that describes a real implementation of this idea. The prototype is made with a smartphone equipped with a Qualcomm Wi-Fi solution. As reported in the original paper: "the design goal was to provide an always-on, highly accurate and low latency pedestrian collision warning system, without introducing significant hardware or processing overhead to the smartphone". The paper states that good performances can be achieved although some problems still need to be solved. Among the others, the accuracy of the position is given by the internal GNSS receiver of the smartphone, the congestion of the wireless medium and the certification of communications and application performances. Indeed, the certification procedure changes a lot depending on whether the application is considered as a supplemental alert or as a complete safety-critical warning system.

A more recent prototype, created by Bosch, addresses motorcyclists is shown in this video [16].

An alternative approach is to exploit the cellular communication channel, owing to its complete availability on mobile devices. Waiting for the complete definition of LTE-V2X and 5G, several papers explored this idea showing good performances. In [17] this approach is theoretically described, also taking in account the road-safety system in terms of energy consumption (on the smartphone). In other papers, LTE communications is used, together with Wi-Fi, to exploit the advantages of both channels, i.e. the more extended communication range of LTE and the low-delays of Wi-Fi direct communication.

A further example is a study on the use of a pure Wi-Fi solution [18] that demonstrates the possibility of effectively using such a channel for the safety purposes.

Finally, a different vision is to use different radio systems, with a dedicated transmitting device carried out by the VRU. The same system can also be used to compute the distance and the position of the users without the need of a GNSS device and the related issues (e.g. accuracy in urban areas). One of the most important works in this sense is done by the Ko-TAG [15] project that uses an RFID-like approach.

1.1.6 French pilot site

In the French pilot site, there is no direct communication between vehicles and pedestrians. Similar to the bicycles, pedestrians transmit their CAM- and DENM-like messages to the IoT platform via 4G/LTE channels from their smart phones. These messages feed the risk assessments algorithm with the appropriate information in order to estimate potential dangerous situations. The result of the algorithm will be transferred to the pedestrian's smart phone via 4G/LTE communications.

1.1.7 Dutch pilot site

In the Brainport Urban Driving use case, a smartphone application is developed which connects to the OneM2M IoT platform. This smartphone app uses Global Positioning System (GPS) localisation to localise the VRUs on campus. This information is used to inform the vehicle of a possible VRU on the road where the vehicle is also driving. The vehicle has to adapt its speed accordingly. Also, the other way around, the vehicle is sending its location to the smartphone using OneM2M, in order to warn VRU of an automated driving vehicle approaching. Second to the smartphone, ITS-G5 beacons are also used to correlate the data transmitted from the smartphone with the location transmitted from the ITS-G5 beacons.

Finally, a Wi-Fi sniffer is used to detect surrounding Wi-Fi enabled devices, which can be used to detect crowdedness by detection of pedestrians and cyclists on campus, using their smartphone or other devices as trackers. While Wi-Fi detection applies to smartphones in the vicinity based on Wi-Fi sensing range (i.e. about 30 meters in outdoor scenarios), filtering mechanisms based on RSSI levels may be used to detect only pedestrians closer to the vehicle. Due to the relatively low position accuracy using this technology, the output (number of devices detected and location of detection, logged by GPS) will only be used to map a crowdedness mapping of the campus and not to individually position VRUs with smartphones. This information will then be used to inform other AD vehicle on how many VRUs are on a certain road, so they can adequately decide to take the less crowded routes.

1.1.8 Italian pilot site

In the Italian PS, V2P is not implemented. In the urban use case, VRUs are detected through a camera and a notification is sent to the cars using a DENM message.

1.1.9 Spanish pilot site

In the Spanish Pilot Site, there are no such hardware or software components that provide this kind of communication.

1.1.10 Finnish pilot site

In the Finnish pilot site, V2P is not implemented. In the urban driving use case VRUs are detected through cameras at the infrastructure.

2.3 Vehicle to device (V2D) domain

1.1.11 French pilot site

The car sharing and car rebalancing use cases in Versailles rely on a mobile application that allows the user to unlock and start one's car. The system is provided by Kuantic and consists of an on-board unit that communicates with the mobile application via Bluetooth Low Energy (BLE).

An embedded interface is also developed to display information about the car to the driver. It consists of an Android tablet that communicates with the car through a serial and an Ethernet link. The serial link is used to communicate with low-level network in the car and display failures, etc. The Ethernet one is used to communicate with the AD units and, for example, to guide the driver during the switch between manual and autonomous driving and autonomous and manual driving.

During the car sharing use case, this interface displays its position on a map to the user. This position comes from the car GPS through the Ethernet link. It also displays an alert when the vehicle enters a zone where the autonomous driving is allowed and when the vehicle is approaching the end of this zone. In the car rebalancing use case, this interface is used to display the state of each car in the convoy.

1.1.12 Dutch pilot site

In the Platooning use case, the drivers are notified in their vehicle by the platoon manager about the platoon status and related information. It uses an existing HMI interface (screen centred on dashboard), which has been slightly modified for this purpose. Additionally, the lead driver is informed in an intuitive way about speed and lane advice via a dedicated app that runs on a smart phone or tablet device.

In the automated valet parking (AVP) use case, the vehicle communicates with a smartphone device using IoT platform over 5G/LTE communication network. The AVP App running on the smartphone receives information such as vehicle state (e.g. current vehicle position, current AVP action and phase) and is able to send commands like “park” or “collect” to the vehicle. The detailed data to be exchanged over IoT platform has been defined by the DMAG (data modelling group).

1.1.13 Italian pilot site

Starting from mature standardised technologies from the fields of IoT, automotive and cooperative ITS, a general purpose platform has been proposed capable of delivering non-safety critical services to a set of final users including AD cars (see Figure 3) [20].

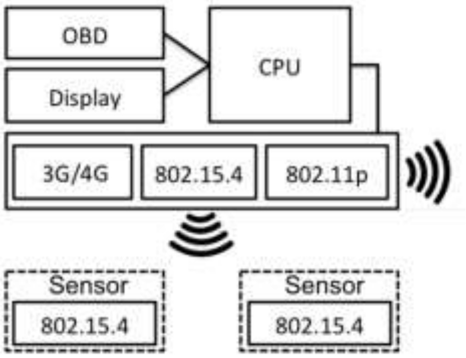

Figure 3 – Architecture of the V2D bridge connecting to a WSN

The vehicular IoT bridge should enable bidirectional semantic-full communications between vehicles and application entities both in-vehicle and in road side infrastructure nodes. The bridge contains a processing unit able to manage all communication interfaces: an IEEE 802.15.4 wireless interface, an OBD/CAN interface, the IEEE 802.11p (ETSI ITS-G5) transceiver and the 3G/4G modem for cellular communication. At the network layer, the bridge should be able to address 6LoWPAN destinations (in order to talk with the on board WSN) and to address other C-ITS station using the GeoNetworking protocol. From outside, the bridge should be addressable as an IP node.

The bridge should, at least, handle at the transport layer UDP (User Datagram Protocol) and BTP (Basic Transport Protocol) [13] and, at the application layer, CoAP communications. We also require for a bridge to abstract all the on-board generated data. This involves the abstraction of all the data that are shared on the OBD/CAN network. Therefore, it should be able to read the main messages in accordance with OBDII standards and to aggregate them with the information coming from wireless sensor network.

In the Italian PS use cases, the IoT bridge is the OBU developed by ISMB that can manage several different devices (V2D): it will manage the connection to a tablet that will be used as HMI and as a sensor for vibration data. The OBU will also interface with an IMU via CAN and with a 6LoWPAN dedicated vibration sensor.

1.1.14 Spanish pilot site

In the Spanish Pilot Site, there is not direct V2D communication implemented. Every possible device which information may be needed by the vehicle will be received through the IoT infrastructure, being this communication then dependant from the V2I domain.

1.1.15 Finnish pilot site

In the Finnish Pilot site, there is no directed V2D communication implemented. A vehicle HMI is integrated in the vehicle. The content shown depends on the use case.

2.4 Vehicle to grid (V2G) domain

Vehicle to grid domain is not addressed in the use cases developed in AUTOPILOT.

2.5 Vehicle to infrastructure (V2I) domain

1.1.16 French pilot site

Within the platooning use case in the city centre of Versailles, the platoon has to pass through two complicated cross-roads. To do so, it is necessary to have vehicle-to-infrastructure communication.

When the platoon is approaching, the complicated intersection, the RSU detects the lead vehicle and passes the message on to the traffic light controller for it to change its phase in order to give the priority to the platoon. The traffic lights interrupt their usual phase and switch specific traffic lights to green/red so that the platoon can cross safely. Once the RSU has communicated with the AKKA cloud through the OneM2M server, the OBU is informed on whether or not the platoon can continue following its route. Once the platoon has gone past the junction, it goes back into its classic functioning mode.

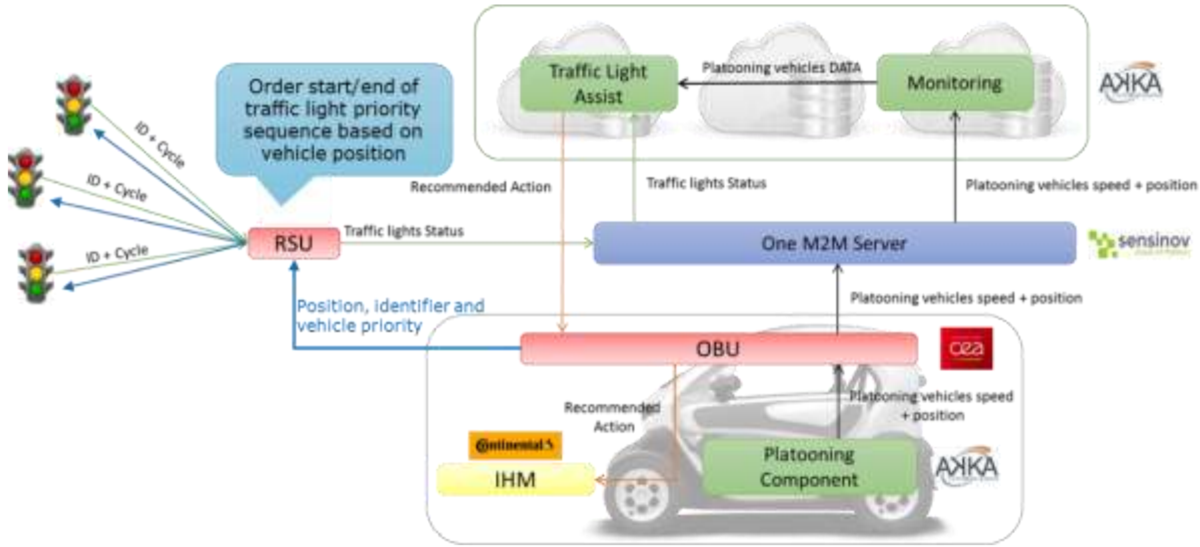

Figure 4 - Traffic light assist architecture for platooning in Versailles (complicated crossroads)

1.1.17 Dutch pilot site

Highway:

All exchanges from/to vehicles go through the infrastructure as depicted below. There are four major components of the system: the Detection (of anomalies by leading ego vehicles), the Reporting (of anomalies to the Cloud), the Validation (or learning of hazards presence) and the Information (for the control of following vehicles).

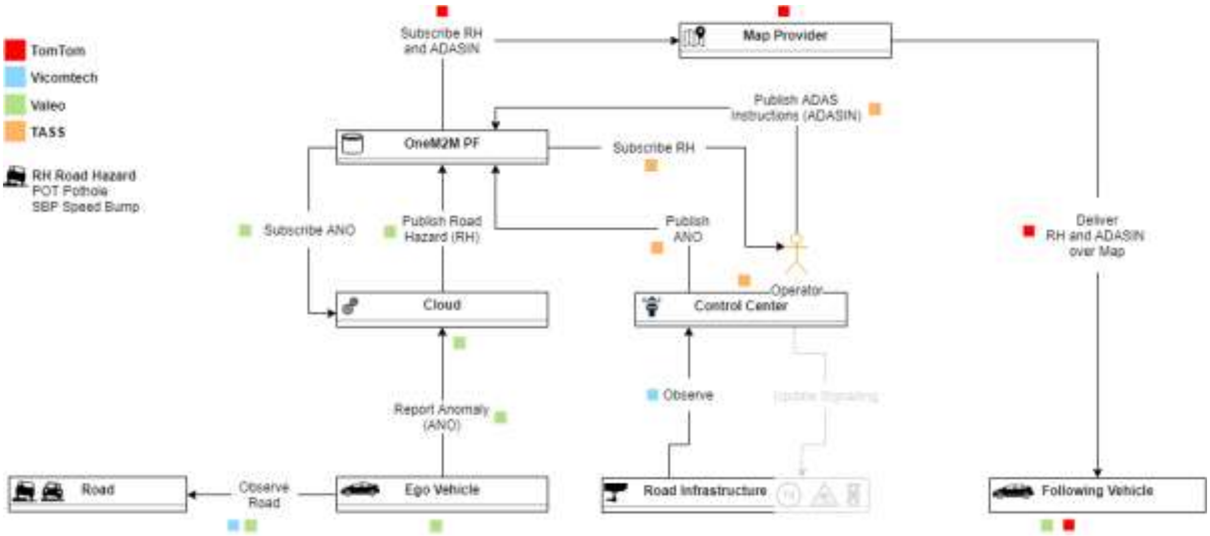

Figure 5 - System view of Highway Pilot in Brainport

Only the reporting and information components rely on V2I communication:

- For Reporting, the vehicle communicates with the Cloud with MQTT and HTTP over a 4G connection.
- For Information, the vehicle communicates with the Map Provider with WebSocket and HTTP over a 4G connection.

Platooning:

Platooning use case uses V2I communication in three different ways:

- Broadcasting ITS-G5 CAM messages that are intercepted by the TASS instrumented facility along the A270 to support vehicle detection
- Exchanging platoon status information with the cloud-based Platoon Service that involves IoT (oneM2M) and cellular (commercial 4G) technology.
- Publishing data to and retrieving data from an IoT-enabled (oneM2M) Local Dynamic Map service deployed at the roadside (A270). This concerns data that can be used to increase the environmental perception of participating vehicles (platoon vehicles and others). This functionality is still under development at the time of writing. The communication channel will be realized through the Hi-5 pre-5G network (TNO), which provides coverage over a part of the A270 (1 base station).
- Traffic lights: Status information of four traffic light controllers controlled by third party Dynniq (RSU701, RSU804, RSU805 and RSU806; one on each successive junction on the N270/Europaweg, Helmond) are received by the TASS MQTT clients from the MQTT broker provided by Dynniq. The data in binary format is converted to JSON format with the ASN.1 decoders of TNO and published to the respective containers on the OneM2M platform. The binary data is also published to the OneM2M MQTT broker. All services and vehicles subscribed to this service can pick up this data. The TLCs all operate in the traffic adaptive mode, which means the cycle changes when a vehicle (road sensor) or pedestrian (button

controlled) approaches the junction.

Automated valet parking:

- Parking spot occupancy detection and obstacle detection: The AVP use case features stationary roadside camera and the micro aerial vehicle (MAV) as infrastructure devices. The MAV and the camera detect free parking spots and obstacles and send this information to the vehicle via the AVP parking management service (PMS) app. The vehicle communicates with the infrastructure devices using the IoT platform over the cellular network connectivity (e.g. 5G/LTE). The MAV is managed by DLR and the roadside camera by TASS and Vicomtech.
- Micro aerial vehicle: the MAV detects the free parking spots and obstacle, processes the data and publishes the parking spot and obstacle status information to the IBM Watson IoT platform. The Parking Management Service App from DLR as IoT application registered by the Watson IoT platform receives this data over MQTT and publishes it to the AVP vehicle.
- Roadside stationary Camera: TASS is providing parking spot status and obstacle status update information. Vicomtech deep learning algorithms send out parking spot status and obstacle status (along the access road to the parking lot). These algorithms are running in the TASS servers and use Advanced Message Queuing Protocol to communicate with TASS Parking Spot Entity, which then publishes them to the containers in the OneM2M platform. TASS also formats the data to a Watson-specific format and publishes them. TNO/TASS vehicle subscribed to this information gets these updates from the OneM2M platform. The data in the Watson-specific format is subscribed by an interworking proxy, which then forwards it to the IBM Watson platform. DLR vehicle or the Parking Management Service App from DLR receives this data from Watson IoT platform over MQTT. For evaluation purposes, TASS Parking spot entity also forwards the data directly to IBM Watson platform. The data flow diagram is shown in Figure 6. As the AVP camera and network planned to be installed in the Automotive Campus 10 building and its parking lot got delayed, TASS has set up two temporary cameras overlooking the TASS parking lot and the access road to it. These are added to the TASS test-site facilities and are accessible like the road side camera as seen in Figure 6. AVP data models follow the SENSORIS and DATEX data model, which is currently being standardized (for AUTOPILOT community) in the data modelling group.

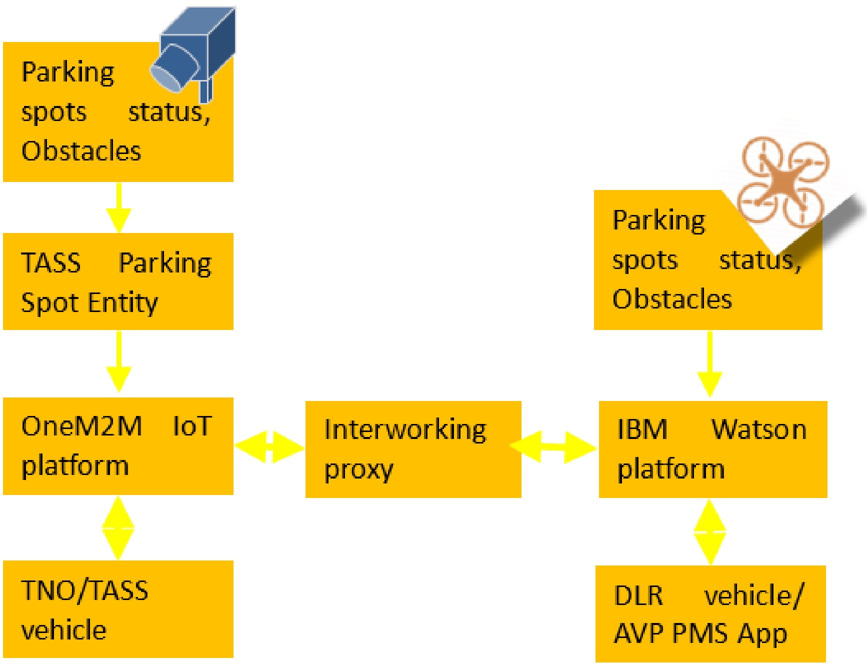

Figure 6 - Interaction between AVP devices and IoT platforms

1.1.18 Italian pilot site

Vehicle to Infrastructure communications are managed using two different channels. The first one is the classic DSRC that will be used to exchange Decentralized Environmental Notification Messages (DENM) with the RSUs and SPAT/MAP messages with the traffic lights. The second channel is LTE. This will be mainly used to send information to the OneM2M platform.

In the Italian PS, DENM are used to notify alert sensed by IoT devices. More in details:

Highway:

- Puddle detection: in the PS, some dedicated 6LoWPAN sensors will detect a puddle. The sensors are connected to an RSU that sends the information directly to the surrounding cars using a DENM message. The same information will also be sent to the OneM2M platform (via fixed network) and then, thanks to Continental and CRF cloud, it is validated and sent to the relevant vehicles, using the LTE channel. The information is sent in the form of a different speed limit for the portion of highway affected by the puddles. In this way both short and long-range communication are covered;
- Road works: the road works notification is sent in the same way described above using an RSU to notify the vehicles. The road works can be fixed or mobile.

Urban:

- Pedestrian red-light violation: in this use case, pedestrians are detected thanks to a smart camera. The information is combined with the status of the pedestrian lights and in case of violation, a message is sent using a DENM notification. The message is also sent to the OneM2M platform;
- Fallen bicycle: ISMB will provide a bicycle equipped with an IoT in-vehicle platform. This device will be equipped with sensors that permit to detect when the bicycle has fallen. This information is automatically sent via DENM by the bicycle. If the message is received by an RSU, this will be sent to the OneM2M platform.

Highway and urban:

- Pothole detection: the in-vehicle platform will act as a “virtual sensor” for vibration. The information can be taken by 1) a 6LoWPAN sensor, 2) a smartphone/tablet or 3) an Inertial Measurement Unit (IMU). The virtual sensor can work with only one source of information or combines different sources. When a pothole is detected, a message is sent to the OneM2M platform where it is available for all the other vehicles.

V2I communications are used to report relevant information coming from IoT to the OneM2M platform. These data are then used to give useful feedback to the autonomous driving function.

1.1.19 Spanish pilot site

Vehicle to Infrastructure communications are supported both with cellular network connectivity and Wi-Fi. Through these channels, the bidirectional IoT communication will be performed, sending and receiving messages following the oneM2M standard.

In the different use cases carried out in the Spanish PS, the communication is as follows:

Urban:

- Traffic Lights: in the pilot site, the different involved traffic lights will be connected to RSU. These RSU will be monitoring the status of the traffic lights and publishing it to the IoT cloud platform (IBM Watson). These statuses will be obtained by the in-vehicle IoT platform through an Urban Server, which will be providing and filtering this information to any connected vehicle.

- Vulnerable Road Users: in order to detect VRUs, a smart camera will be used, located in the surroundings of the road. This camera will detect any pedestrian located in a relevant area and send a VRU event message to the IoT cloud platform (IBM Watson). Afterwards, this information will be collected by the mentioned Urban Server, which will provide and filter it to any connected vehicle.
- Hazards: in order to obtain the different hazard events that might occur, the control management system of the public authorities will be used. By using a module that obtains the different hazard events and translates them to IoT messages, publishing them to the Watson IoT platform, these hazards are available to any vehicle connected to the same Urban Server mentioned above.

Automated Valet Parking:

- Drop-off and pickup: in the parking provided by the Vigo City Council, a parking management system is developed. This parking management system can forward the user's command of pickup and drop-off to the vehicle. Also, this system can detect VRU that afterwards would be published to the IoT platform in the same way that it is done for the Urban VRU. The in-vehicle platform can then receive these commands and VRU events adapting its behaviour.

V2I communications are used to report relevant information coming from the IoT platform. This data is then used to give useful feedback to the autonomous driving functions.

1.1.20 Finnish pilot site

In the Finnish pilot site, Vehicle to Infrastructure communications are supported both with cellular 4G/LTE communications and with ITS-G5.

In the different use cases in the Finnish pilot site, the communication is as follows:

Urban Driving:

- Traffic Lights: in the pilot site, real-time information on signal state and the next phase is available both through ITS-G5 as standard SPAT/MAP messages and through cellular communications, through connection to the traffic light operator's (Dynniq) server.
- Vulnerable Road Users: in order to detect VRUs, a smart camera will be used, which is installed at the mobile road side unit of VTT. This camera will detect pedestrians and cyclists located in a relevant area and send a VRU event message to the IoT cloud platform. From there, the information will be made available to the vehicle.

Automated Valet Parking:

- Traffic cameras, installed at VTT's mobile road side unit, monitor the parking and detect objects and pedestrians either at the parking spaces or on the potential vehicle paths, and send the information to the IoT platform.

3. IoT sensors and actuators for autonomous vehicle applications

Sensors and actuators are essential parts of the automated vehicle IoT ecosystem and are used to inform and warn the driver, or even actively interfere in the driving.

Automated vehicle systems consist of inputs from a large variety of sensors, data signal condition and decision making by central or edge processing units and outputs to a large variety of actuators.

The integration of these IoT devices into the IoT ecosystem includes software operating systems, interfaces, gateways and communication capabilities as illustrated in Figure 7.

The target functionalities may be solved by the use of different or similar types of sensors and actuators. Increased functionality and in particular increased reliability can be achieved by technology redundancy through sensor fusion (multiple sensors and multiple functions).

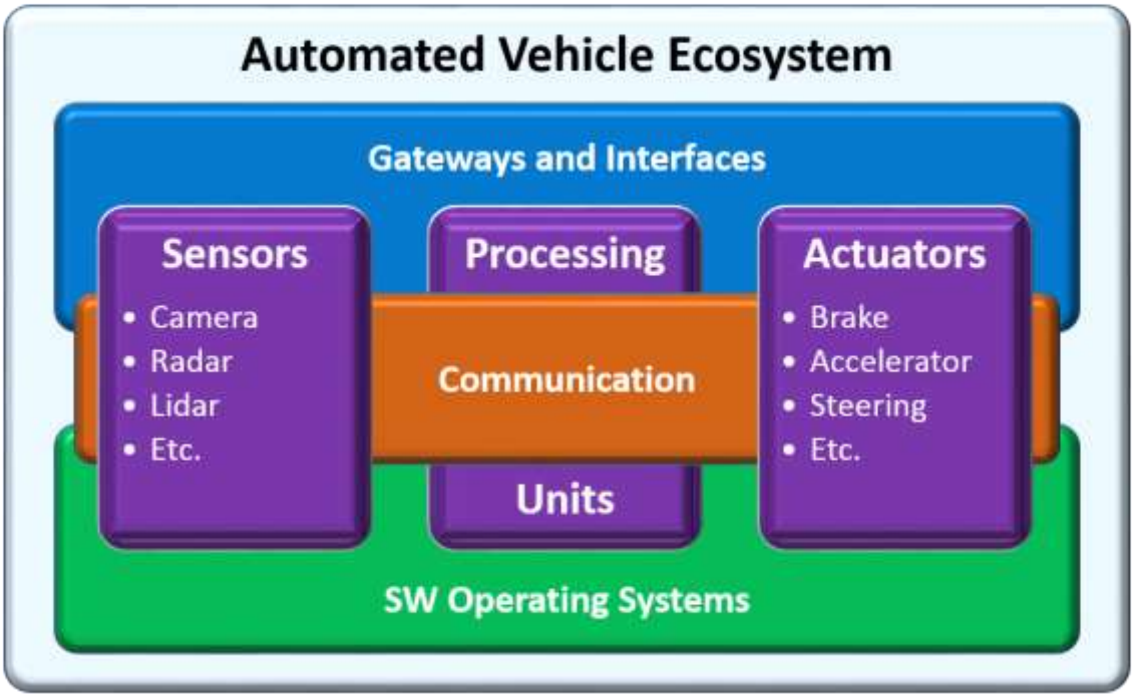

Figure 7 – Automated vehicle ecosystem

The functions covered by the sensors and actuators are active cruise control (ACC), lane departure warning system (LDWS), lane keep assist (LKA), park assist (PA), automatic emergency braking (AEB), driver monitoring (DM), automatic pilot (AP), weather monitoring, car sharing, car parking, environment monitoring, road state and crowd monitoring as illustrated in Figure 8.

Chapter 3 describes the IoT devices, sensors and actuators used and/or developed by AUTOPILOT partners in the different use cases, which are integrated into the IoT ecosystem for the autonomous vehicle applications functions (the use cases and functions are described further in Chapter 4).

In order to give a better overview of the sensor/actuators that support the autonomous driving, Chapter 3 includes devices that are integrated in the vehicle to provide autonomous driving functions but are not part of the use case demonstrated.

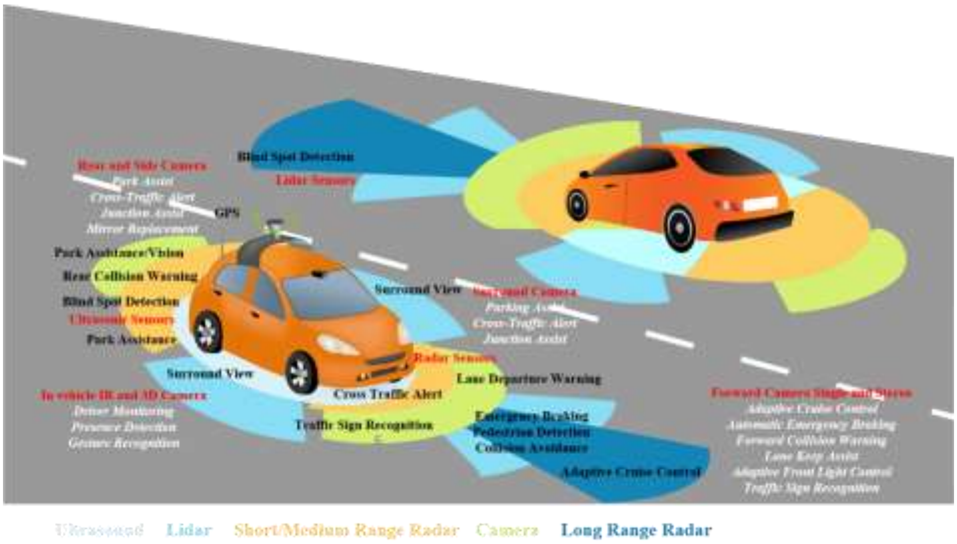

Figure 8 – Sensors, actuators and functions

Table 1 - Overview about the IoT sensors used in different pilot sites

	Pilot sites (PS)				
IoT Sensors	**Tampere**	**Versailles**	**Livorno**	**Brainport**	**Vigo**
Ultrasonic					X
Radar (Medium-range radar - MRR)					X
Optical (Long-distance Camera)					X
Optical (Smart camera)	X				X
Optical (Stereo camera)			X	X	
Optical (Panoramic camera)				X	
Optical (Mono sensor camera)				X	
Lidar	X		X	X	X
Accelerometer			X	X	
UWB Localisation	X			X	
BLE Beacons		X		X	
NB-IoT sensors			X		
Crowd detector device with Wi-Fi and GPS sensors				X	
GNSS receiver				X	
LoRaWAN		X			

3.1 Ultrasonic

Ultrasonic sensor technology is typically used in parking-assisted solutions, and for obstacle and pedestrian detection. The ultrasonic technology is not used at the Italian, Dutch and Finnish pilot sites.

1.1.21 Spanish pilot site

In order to detect the parking space and help the parking manoeuvre in the Valet Parking Use Case, the information provided by the own ultrasound sensors of the C4 Picasso is used. The vehicles have 6 ultrasound sensors integrated in the front bumper and six others in the rear.

3.2 Radar

Radar sensors use electromagnetic waves for object detection, and is almost unaffected by temperature, snow, rain, fog, dirt, darkness or changing light conditions [5][6]. It is also possible to mount radar sensors behind electromagnetic transparent material-like bumpers. Radar sensors can detect objects like pedestrians or other vehicles and track their movements, by measuring distances, angles and speed. As illustrated in Figure 9, the automotive radar sensors can be categorized according to observation range; long, medium and short-range radars, and are facilitating different applications [5].

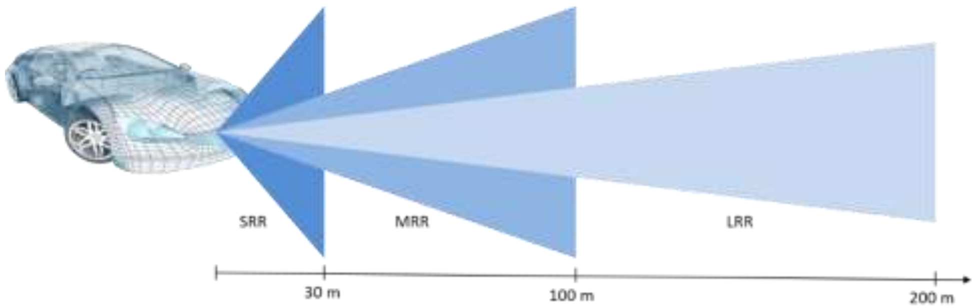

Figure 9 – Radar observation ranges [5]

1.1.22 Long-range radar (LRR)

Long-range radar (LRR) technology is typically used in adaptive cruise control and automated emergency braking solutions.

1.1.23 Medium-range radar (MRR)

Medium-range radar (MRR) technology is typically used for cross traffic detection, blind spot detection and lane change assistance.

1.1.24 Spanish pilot site

The 3 C4 Picassos prototypes integrate 4 radars in the corners of the Vehicle. They are 79GH sensors with a field of view of 150º and a range up to 100m.

Radars will be used in Urban Driving Use case in order to detect objects around the car and fusion this information with that obtained by other sensors.

1.1.25 Short-range radar (SRR)

Short-range radar (SRR) technology is typically used for pedestrian and obstacles detection and in parking assisted solutions.

3.3 Optical

1.1.26 Long distance camera

1.1.27 Spanish pilot site

The vehicles integrate a Mobileye 6 Open Protocol system with the camera in the Windshield. This sensor can detect pedestrians up to 40m and vehicles up to 150m. It is used to position the car in Urban Driving, both laterally (with the road lines) and longitudinally (matching objects with the map). They are also used to detect pedestrians and cars, information which is uploaded to the IoT open platform.

1.1.28 Smart Camera

1.1.29 Spanish pilot site

In the Vigo Pilot Site, some elements of the infrastructure will be equipped with a Smart Camera or a Camera-Processing Module with the behaviour similar to a smart camera. The intention is to inform cars about the presence/absence of objects in pedestrian crossings (crosswalks).

Since the camera will be static, fixed to some infrastructure, the algorithms used to detect objects/pedestrians can be designed for this situation. Background subtraction algorithms are widely used to detect changes in images obtained from static cameras. The algorithms attempt to fit some statistical model to the images without objects and to detect the parts of the image that don´t fit the model when an object is present.

The most common approaches are using codewords and Gaussian mixtures to statistically describe the model. In Gaussian mixtures model, a group of pdf (probability density function) will be used to describe each pixel individually, so changes as small as a pixel can be detected. Changes in the images are processed to classify the pixels in background or foreground. Foreground pixels are post-processed to discard noise or very small objects. Foreground pixels are then upgraded to objects and objects are classified as moving or static. The intention is to detect every object present in the road, not only pedestrians. After an object is detected, we also perform a cascade detector using the HOG algorithm [27] to classify objects in pedestrian/no pedestrian categories. The system detects pedestrians up to a distance of 20 meters from the camera.

The smart camera used in Vigo Pilot Site in order to inform about pedestrians crossing via cooperative communication has the following specifications and features:

Table 2 – Smart camera characteristics

Smart Camera Characteristics	
Manufacturer	IDS Imaging Development Systems GmbH
Model	UI-1221-LE
Interface	USB 2.0
Sensor	CMOS
Sensor Size (px)	752x480
Optical class	1/3”

Pixel size	6 um
Frame rate max	87 fps
Working frame rate	30 fps
Lens focal distance	3.6 mm
Detection distance	3-16 m

Figure 10 – Smart camera (UI-1221-LE, IDS GmbH)

1.1.30 Finnish pilot site

The city of Tampere is installing traffic cameras at major intersections. The camera to be used at the Finnish pilot site is the same camera type, HIKVISION DS-2DF8223I-AEL.

Figure 11 - HIKVISION DS-2DF8223I-AEL camera used in the Finnish pilot site

The camera is installed on the mobile Road side unit from VTT, allowing it to be used for both use cases. The camera is used in a fixed position, and areas of interest are defined within the view. For the detection, the Jetson development board is used (see also Figure 13 and Table 4). The detection algorithm is based on the YOLO real-time object detection system.

1.1.31 Stereo camera

1.1.32 Italian pilot site

Camera is one of the main sensors used in today's autonomous driving applications. The market already sells several mature products both directly integrated by the OEMs and aftermarket. These products permit to detect different obstacles, like pedestrians, other cars, etc., and to estimate their distance and relative speeds. Cameras are also used for other tasks like lane detection, visual odometry, road sign reading applications, etc.

Stereo vision is the reconstruction of a 3D image starting from two different pictures taken by two different cameras separated by a horizontal distance. A stereo-camera consists indeed in two (or more) cameras that are typically separated by the distance of human eyes (e.g. the average intra-ocular distance is about 6 cm). Indeed, the concept at the basis of stereo vision is the same of human vision. The two images must be compared to obtain information about the relative depth of the images. It is indeed possible to determine the relative distance of the detected objects in the image thanks to two different prospective of the same scene. Main issue of stereo vision system is to identify the objects in each image.

Stereo-camera can provide accurate distance measurements in the near range, while they have to exploit additional information from the context and the environment to provide distance measurements for far objects. Stereo vision is often used for obstacles detection, because it can give a three-dimensional representation of the scene. The type of processing needed to build the 3D image is very costly from a computational point of view but gives better results than mono vision systems. However, the 3D reconstruction is not as precise as the one provided by active perception sensors, such as radars or LiDARs, and resulting 3D images can be affected by some noise.

Stereo vision is widely used in several field of application like autonomous system, robotics, manufacturing, entertainment, etc.

Notably, in the autonomous system arena, the main actors are using stereo-camera to sense the world around the vehicle to have a 3D representation of the surrounding environment and to recognize specific target objects. For example, Google, Tesla, BMW and Mercedes-Benz are using this technology for several tasks like pedestrian detection, road surface-analysis, lane detection, etc.

In the last years, there was a lot of research about the analysis of stereo-images. Today standard algorithms are available also in popular open-source libraries like OpenCV. The main fields of research are nowadays focused on improving the quality of recognition. Deep learning (a branch of machine learning) is one of the most explored techniques to improve the quality of detection done by a camera. It involves the training of a computer program with a large number of images. The objective is to teach the program how to do the task by itself. Deep learning mimics the way of working of the human brain to solve complex problems. This implies that the program must take in input a large number of images, i.e. very different situations to cope with more or less every possible thing that can happen on the street.

Crowdsourcing, that is a data collection done by a user of a certain service, is typically used to collect this wide number of different situations (as images) used to train the deep learning programs.

The evolution of stereo-vision is 360° cameras that give a complete view of the surrounding producing a 2D or 3D image. For 3D realistic image, these techniques should use from 4 to 6 cameras, further increasing the computational complexity of the involved algorithms.

The importance of pedestrian detection for road safety has led to the development and availability of commercial pedestrian detection systems. While these systems are able to detect the presence of pedestrian(s) on the road and some even provide the number of pedestrians that they can detect, the cost of such systems is normally very high.

As mentioned earlier there are different methods which can be deployed to detect pedestrians and finally, also measure their respective distances. Each method has its own associated parameters that can help to select the right pedestrian detection methodology for the Italian Pilot sites.

Considering the various different options, the availability of their hardware and software components and other parameters such as the cost, accuracy and the development in this field, a pedestrian detection system based on a stereo vision camera emerges to be a successful candidate. In particular, for the Italian Pilot Sites, the stereo vision camera system ZED from Stereolabs has a relatively low cost, as compared to other alternative solutions, and is able to provide 3D depth perception and motion tracking features specifically related to pedestrian detection. Table 3 lists the specifications of the stereo camera.

Figure 12 – ZED camera from Stereolabs

Table 3 – ZED Stereo Camera Specifications

ZED Stereo Camera Specifications	
Depth perception range	0.5 to 20 meters
Focus range	6-DoF positional tracking
Video Modes	WVGA – 2.2K
Frame Rate	15-100 (depends on video mode)
Depth Resolution	Equivalent to video resolution (video mode)
Motion	6-axis pose accuracy with SLAM and Real-time depth-based visual odometry
Lens	Field of View: 110° (D) max.
Lens aperture	F 2.0
Sensor size	1/3" backside illumination
Sensor Resolution	4M pixels per sensor with large 2-micron pixels
Camera controls	Adjust Resolution, Frame-rate, Exposure, Brightness, Contrast, Saturation, Gamma and White Balance
Connectivity	USB 3.0 port
Power	Power via USB 5V / 380mA
Operating Temperature	0°C to +45°C
Dimensions	175 x 30 x 33 mm (6.89 x 1.18 x 1.3'')
Weight	159 g (0.35 lb)
Os compatibility	Linux , Windows
Development Environment	An SDK is provided with integrated support of openCV

The stereo camera provides all the necessary features to implement pedestrian detection, however in order to use these features, and particularly, the depth perception feature of the camera, a GPU with high processing power is required. In general, real-time processing of videos for such applications, does however, require high computational capabilities.

To implement the pedestrian detection framework for the Italian pilot sites, the embedded development board Jetson Tx2 by Nvidia would be used. The Nvidia Jetson TX2 is a development board that provides a high grade Nvidia GPU with 256 CUDA cores. Table 4 lists the specifications of the Jetson TX2 board.

Figure 13 – Nvidia Jetson TX2 Development Board

Table 4 – Nvidia Jetson TX2 Development Board Specifications

Nvidia Jetson TX2 Specifications	
GPU	Nvidia Pascal, 256 CUDA cores
CPU	HMP Dual Denver 2/2 MB L2 + Quad ARM® A57/2 MB L2
Video	4K x 2K 60 Hz Encode (HEVC) 4K x 2K 60 Hz Decode (12-Bit Support)
Memory	8 GB 128 bit LPDDR4 59.7 GB/s
Data Storage	32 GB eMMC, SDIO, SATA
USB	USB 3.0 + USB 2.0
Connectivity	1 Gigabit Ethernet, IEEE 802.11ac WLAN, Bluetooth
Interfaces	Full-Size SD, USB 2.0 Micro AB, SATA Data and Power, HDMI, GPIOs, I2C, I2S, SPI, M.2 Key E, TTL UART with Flow Control, PCI-E x4, Display Expansion Header, Camera Expansion Header

1.1.33 Dutch pilot site

In the Dutch pilot site, the stereo camera is used by the micro aerial vehicle (MAV) and by the prototype vehicle. For the valet parking use case, the panoramic camera and mono sensor road site infrastructure camera are used for parking spot occupancy detection and obstacle detection.

Stereo camera used by the MAV:

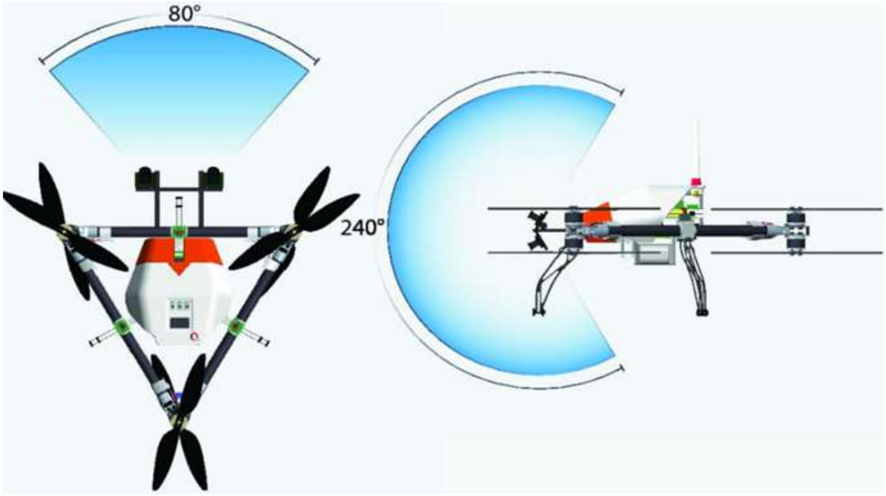

Figure 14 – MAV camera setup. Blue illustrates the field of view of the stereo cameras

The MAV used in the AUTOPILOT project is using two stereo camera systems, which consist of 4 wide-angle cameras. The camera setup provides an approximated 240° vertical field of view and an 80° horizontal field of view as illustrated in Figure 14. Therefore, the MAV can perceive objects that are under it and above it to the same time. The MAV can create a full-spherical 3D map whenever it has rotated around its z-axis. The arrangement of the cameras and the separate control of them make the system well suited for outdoor scenes with high dynamic range situations. The cameras are used for navigation and mission purposes.

Stereo camera used by the prototype vehicle:

For the Brainport, the vehicle of TU/e uses a custom-built stereo camera built on basis of two Sekonix cameras with an ONSEMI CMOS Image Sensor AR023. This stereo camera is used to estimate the depth (disparity estimation) and together with VICOMTECH's VRU detection algorithm, it is capable of detecting, classifying and locating VRU's in front of the vehicle.

Stereo vision makes it possible to estimate 3D scene geometry given only two images from the same scene. In this case, we consider a conventional stereo rig in which two cameras are separated by a horizontal baseline. Based on epipolar geometry, a new image representing depth estimations can be calculated from the two images of the cameras. This is usually called "disparity map" and it measures the horizontal difference in pixels between small image patches that belong to the same object in both (left and right) images of the stereo pair. The corresponding patches are projections of the same 3D point/area in the right and left images. A stereo calibration procedure is required to calculate a set of intrinsic camera parameters and extrinsic ones related to the relative position of the cameras as a pair.

Using the disparity map and calibrated parameters, an estimate of the 3D location of objects in the scene can be obtained. In this procedure, there are two main sources of errors: pointing errors due to calibration inaccuracies and matching errors in the algorithm that finds corresponding patches on both images.

The uncertainty error grows quadratically with respect to the depth. Thus, the farther the object, the higher uncertainty in its 3D location with respect to the cameras position is. Besides, partial occlusions (parts visible in one camera and not seen on the other) and the chosen algorithm for stereo matching, impact the resulting disparity map that can be very sparse with areas of unknown disparity.

The VRU module (developed by Vicomtech with TU/e) will integrate disparity data in the following way: given a 2D detection of a person (bounding box in image) in front of the vehicle that has the stereo camera set installed, the (x,y) coordinates of the centre of this box can be transformed into a 3D location (X,Y,Z) relative to the camera set position in the car.

To filter possible errors of the disparity map at location (x,y), the neighbouring pixels belonging to the detected person can be used to aggregate the disparity values towards obtaining a representative mean or median as depth estimation.

With regard to the automated detection of people in images captured by cameras on-board vehicles, Deep Learning approaches have outperformed the classical detection and classification methods in the area of Computer Vision. Several detectors are being tested and the highest performance has been obtained with the mentioned Deep Learning models. Deep Learning is a fast-paced area of research in exponential growth mainly due to its broad success and applicability for different machine learning tasks.

There exist several neural networks architectures, modifications and trained models in the state-of-the-art. Two state-of-the-art frameworks are under tests: Caffee and Tensorflow. To date (plugfest2, mid-May 18), Vicomtech has developed the detector for VRUs in images and this module has been successfully integrated in TU/e vehicle.

In particular, a multi-class object detection model based on Tensorflow has been tested on images provided by TU/e. The estimated 'hit ratio' (=correct detections) for Vulnerable Road Users is above 85% and it can take less than 60ms per frame when employing a computer with a powerful GPU. The implementation of disparity calculation and the estimation of the distance to detected pedestrians are ongoing. The expected distance range from the vehicle will be estimated using the calibration of the cameras and evaluation tests.

Panoramic and mono sensor road site stationary camera:

To support the automated valet parking (AVP) use case, road side stationary cameras are installed in the area where the autonomous vehicles are driving and parking. The cameras are located at the drop up or pickup position of AVP located at different positions: the entrance of the automotive campus building in Brainport, the lanes/access roads to the parking spot and the parking spot

The following cameras are used in the AVP use case:

- 4 x AXIS Q3709-PVE Network Camera
- 3 x AXIS Mono P3227-LVE Camera
- 1 x AXIS P3375-VE Camera

The video streams from these cameras are processed by algorithm running in the back-office servers. The algorithm detects parking spot occupancy and also obstacle along the access road to the parking spot (see Figure 15)

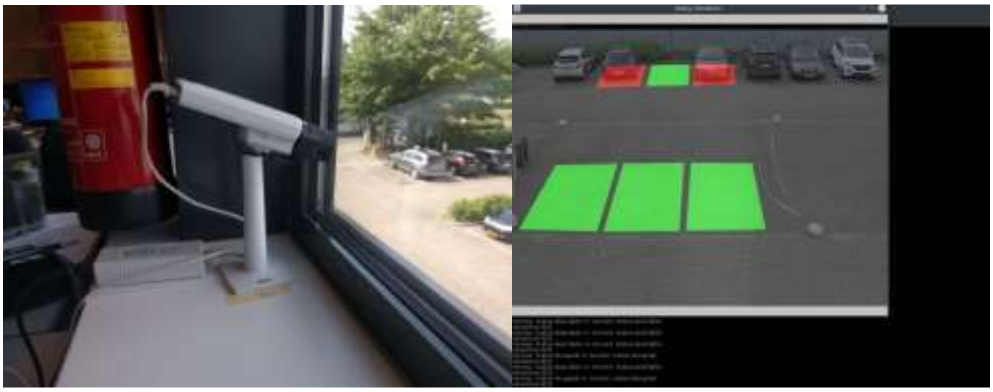

Figure 15 - AVP road unit camera and parking lot detection visualization

3.4 Lidar

The light detection and range (Lidar) technology is typically used for 3D mapping of the surroundings. Among the three primary technology types, the Mechanical mechanism Lidar is the oldest and most traditional technology, the MEMS Lidar is in the beginning development stage as a low-cost solution for low-level automotive safety, while the solid-state hybrid Lidar has been tested for autonomous safety over the years and the costs are decreasing [7]. The solid-state Lidar technology simplifies the previous complex mechanical systems, and enables faster data capture in 3D, capturing pictures instantaneously at real-time speeds.

1.1.34 Spanish pilot site

The Spanish prototypes have two different types of Lidars. One 2D, integrated in the front bumper, which is used to detect other vehicles and perform vehicle following; and another 3D, integrated in the roof, which is used to detect objects and pedestrians around the vehicle in Urban Driving, and to position the vehicle and detect spots in Valet Parking. The 2D Lidar is a Scala 1 from Valeo with a field of view of 145º and a range up to 200m. The 3D one is a VLP-16 Lidar from Velodyne with a field of view of 360º horizontal and 30º vertical, and a range up to 100m.

1.1.35 Dutch pilot site

The Dutch pilot site hosts automotive LiDAR. The connected and automated prototype vehicles used by the AVP for example are equipped with LiDAR. The DLR prototype vehicle is equipped with four IBEO Lux laser scanners. They have a field of view of 85° and range of 200 m each. Three of them are equipped on the front / sides of the vehicle and one is equipped on the rear (see Figure 16).

Figure 16 – DLR prototype, position of laser scanners

TNO vehicle has six LiDARs resulting in a 360-degree view around the vehicle (see Figure 17). The Lidars are mounted at the front and rear bumpers of the vehicle; one in the centre and two in each corner looking sideways. The input of the Lidar is a point cloud, containing the distance for each laser-scan to the nearest object in sight.

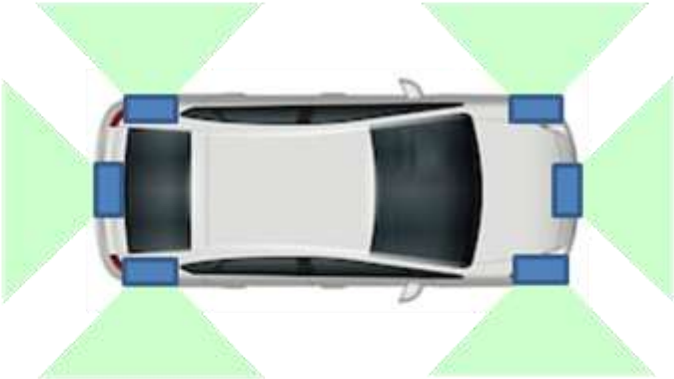

Figure 17 - Position of LiDAR on the TNO vehicle

1.1.36 Italian pilot site

The Livorno Pilot Site hosts two different kinds of LiDAR: an automotive LiDAR and a roadside LiDAR. One of the two connected and automated car prototypes have been equipped with a LiDAR installed at the front bumper of the Jeep Renegade. It is an automotive time-of-flight-based 2D Near-Infrared laser scanner, characterized by a horizontal field of view of about 145°, a detection range of 80 m and a distance resolution smaller than 100 mm. Its main features are listed in Table 5.

This sensor will contribute to perform the obstacle detection task, taking into account the approach already presented in [24] as starting point. It provides as output the list of detected objects with the corresponding parameters (in terms of relative position w.r.t. the Lidar, width, length, orientation, and relative speed). This list is then suitably analysed in order to identify the obstacles that are of interest for the current driving task (e.g. frontal obstacle for Adaptive Cruise Control or Frontal Collision Warning, lateral obstacles for lane change).

Table 5 – Lidar installed in the Jeep Renegade

Main features	
Wavelength of sender IR-LD	905 ± 10 nm
Horizontal FOV	~ 145°
Vertical FOV	3.2° (average)
Horizontal resolution	≥0.25°
Distance resolution	≤100 mm
Range for objects	80 m
Scan rate	25 Hz

The roadside LiDAR prototype is provided by the Industrial Technology Research Institute (ITRI), which joined the project as associated partner. ITRI's LiDAR is road side equipment with a field of view of 360°, which, combined with traffic light and IoT/ITS G5 RSU, enforces the protection of VRUs at intersections (see Figure 18). In fact, the LiDAR will detect real-time information of vehicle and pedestrian (e.g. position. velocity) around the intersection. Then the information is consumed by the

RSU in order to:

1. Calculate time to collision
   - Combining both vehicle and pedestrian information to determine whether there may be a collision. If the time to collision is below the threshold, it will be determined as an imminent accident.
2. Send safety warning message
   - Sending messages to OBU in order to avoid accident
   - Publishing alert message on the oneM2M platform to be consumed by cloud applications

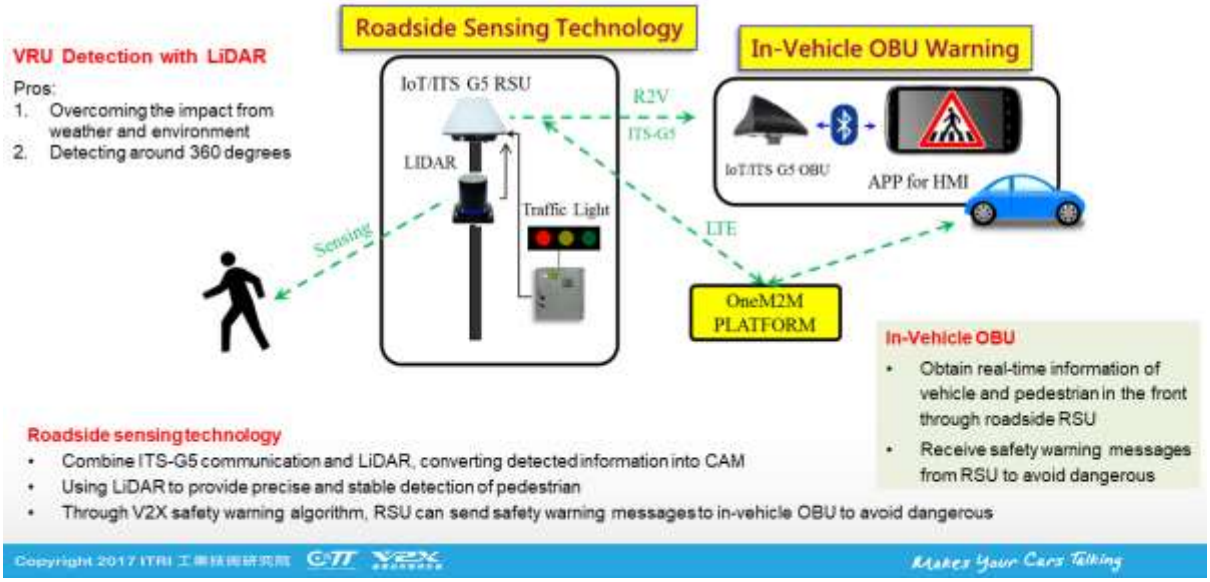

Figure 18 – ITRI prototype, VRU Protection with LIDAR

1.1.37 Finnish pilot site

The Finnish prototype vehicle is equipped with 2 SICK HD LIDARs at the front bumper of the vehicle, and an ibeo LUX LIDAR at the rear of the vehicle. The vehicle is used to detect objects in the vehicle path. The LIDAR data is made available to the other functions in the vehicle.

3.5 Accelerometer

The accelerometer is basically a linear motion sensor, i.e. a device that measures total specific external force of the sensor: it is sensitive to both linear acceleration and the local gravitational field. The basic principle of operation of an accelerometer is the second law of motion. The data from the accelerometer is conventionally reported in units of g (1g= 9.81 m/s^2). In the initial condition and calibration, the accelerometer reports a value of 1g along the z-axis and 0 along the x and y axes when lying at rest face upon a flat table. The gravity vector thus reported is used as a reference for all other linear motion sensing.

With the increasing popularity of smartphones among people, researchers are showing interest in building smart IoT solutions using smartphones because of the embedded sensors, like GPS, accelerometer, gyroscope or magnetometer.

Autonomous vehicles must not only understand vehicle dynamics in terms of position, orientation, direction and velocity of the vehicle, but also whether changes in the relationship between these factors is leading to an unsafe situation either for occupants of the vehicle, bystanders or other road users. The safest ride is the most stable ride.

1.1.38 Dutch pilot site

The prototype vehicles (e.g. TNO, Valeo and DLR vehicle) used in the Dutch pilot site are equipped with accelerometer. An initial measurement unit (IMU) is used to measure the ego-motion of the vehicle. The IMU measures translational accelerations in the three orthogonal directions (forward, lateral and vertical) and it also measures the angular velocities around the three orthonormal axis (roll-, pitch- and yaw-rate). The physical measurement principle is based on the Micro-Electro-Mechanical Systems (MEMS).

1.1.39 Italian pilot site

The Italian Pilot Site (ISMB) has decided to implement the study of a "Virtual Sensor" for pothole detection, using the same approach with three different sensors.

The data of the raw signal accelerations on the 3 axes will be collected and analysed using a Nokia 6 smartphone (https://www.nokia.com/en_my/phones/nokia-6), an inertial 6LoWPAN sensor provided by CNIT and the accelerometer sensor of an inertial measurement unit (IMU) from CRF. The virtual sensor can use one or more acceleration sensors combining the upcoming data in a smart way. The accelerometer measures changes in velocity of the sensor in three dimensions: the linear sensing provides the sensor information about its motion and thus taps, or shakes can be detected. Similarly, orientation can be determined by the sensor's sensitivity to the local gravitational field.

A continuous stream of data related to the linear acceleration of the vehicle on three principal axes, together with the three sets of rotation parameters (pitch, roll and heading), will provide additional measurements related to distance travelled by the autonomous vehicle. This will provide data related to the velocity and the extent of acceleration towards obstructions (see Figure 19 and Figure 20). The data patterns captured by the accelerometer can be used to detect physical activities of the user such as running, walking, and bicycling.

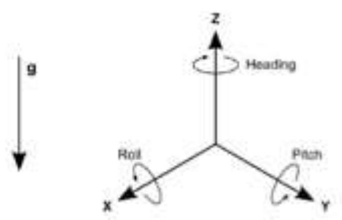

Figure 19 – Gravity vector and heading, pitch and roll about axes

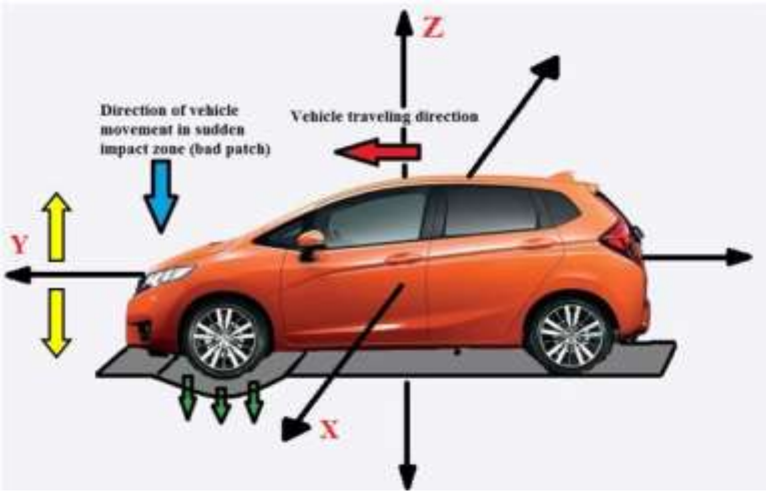

Figure 20 – Downward jerk sensed by accelerometer which occurred due to potholes over the road.

1.1.40 Smartphone Accelerometer

The accelerometer records all vehicle vibrations including vibrations from the engine and the gear box and all swings made by passengers. It is possible to calculate the acceleration of the vehicle in two directions: (1) in the direction of motion of the vehicle to identify the braking; (2) in the direction perpendicular to the direction of motion of the vehicle to identify bumps and potholes. Axes of the phone may not be aligned along the axes of the vehicle. Phone can be in any arbitrary orientation inside the vehicle. To use the accelerometer reading for detecting various events, it is possible to virtually reorient the axes of the smartphone to align along the axes of the vehicle. Readings from the reoriented axes can be used to detect events. Leveraging an accelerometer as a vibration sensor, the characterization of potholes and roads can be done using the readings of the accelerometer.

There are many things that should be taken into consideration to get the accurate reading when reporting the descriptions of this device to the concrete case, when the accelerometer is deployed on the vehicle. When the accelerometer is deployed in a real vehicle, its reading might be biased because the shock absorption system of the vehicle reduces the effect of the potholes. Till now, smartphones in road condition monitoring is limited to recording of accelerations and processing them to discern potholes and monitoring overall condition of road surfaces. Therefore, the data must be pre-processed before it can be used. This can be done for example by using a passband filter. It removes low and high frequencies from the measured data. This makes the data much cleaner and easier to process. Data can also be divided into small segments and normalized to some specific scale to make the feature extraction and classification easier.

Using the mobile device based on mobile sensing techniques to detect potholes, is suitable and convenient: all the motion sensors return multi-dimensional arrays of sensor values for each *SensorEvent*. The linear acceleration sensor provides with a three-dimensional vector representing acceleration along each device axis, excluding gravity [linear acceleration = acceleration - acceleration due to gravity]. The linear acceleration sensor always has an offset, which has to be removed. The simplest way to do this is to build a calibration step into the application, in order to iterate the alignment of the smartphone accelerometer's coordinate system and the vehicle's coordinate system.

Accelerometers use the standard sensor coordinate system. In practice, this means that the following conditions apply when a device is lying flat on a table in its natural orientation (see Figure 21):

- If the device is pushed on the left side (so it moves to the right), the x acceleration value is positive.
- If the device is pushed on the bottom (so it moves away from you), the y acceleration value is positive.
- If the device is pushed toward the sky with an acceleration of A m/s^2, the z acceleration value is equal to A + 9.81, which corresponds to the acceleration of the device (+A m/s^2) minus the force of gravity (-9.81 m/s^2).
- The stationary device will have an acceleration value of +9.81, which corresponds to the acceleration of the device (0 m/s^2 minus the force of gravity, which is -9.81 m/s^2).

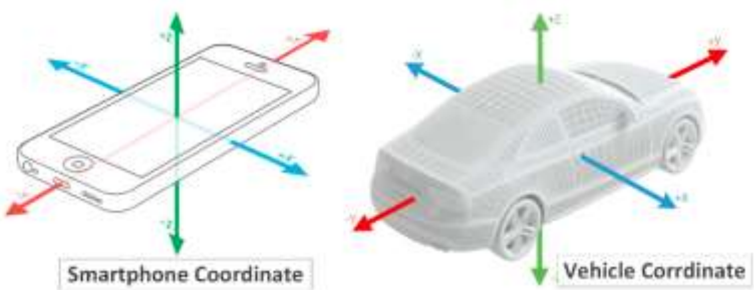

Figure 21 – Coordinate system (relative to a device) that's used by the Sensor API.

1.1.41 6LoWPAN Inertial Sensor

The second sensor used by PS Italy is an inertial unit that supplies raw accelerometer data. This device is supplied by a partner of CNIT (model SensOne). SensOne is a cost-effective wireless device for use in industrial monitoring and ad-hoc sensor network applications. SensOne leverages industry standard IEEE802.15.4 RF protocol for robust and power aware communication interfaces and USB2.0 connectivity. It embeds light, humidity, accelerometer sensors and a low-noise accurate Analog to Digital converter for flexible deployments. The SensOne has been designed for battery-powered Internet of Things applications and natively supports state-of-the-art Internet addressing protocols (e.g. 6LoWPAN), to interoperate seamlessly with other devices.

Table 6 – SensOne inertial sensor characteristics

Inertial Sensor Characteristics		
	Characteristics	**Comment**
Module		
Processor	TI CC2538	ARM Cortex-M3
Frequency	32 MHz	
SRAM	32 KB	16KB Low power
Program Flash	512 KB	
Debug	JTAG	
Battery charger	1-cell	Up to 500mA
RF transceiver		
Frequency Band	2394-2507 MHz	ISM band
Data Rate	250 kbps	
LOS Range	<100 m	
Sensors		
Accelerometer	3-axis	12bit resolution
Humidity	0 – 100% RH	±3% RH
Temperature	-40°C +125°C	±0.4°C
MCU Temperature	0°C +80°C	±0.5°C
Battery level	0 - Full Charge	±1%
Expansions		
Analog input	4	24 bit ADC
I/Os	1	
USB2.0	6	Up to 12Mbit/s
Other interfaces	I2C / UART / SPI	1 / 2 / 2
Electromechanical		
Power supply	3.3 – 5 V	DC/DC converter

Consumption	40mA	
Battery	3500mAh	
Size	48x48 mm	

The triple-axis MEMS accelerometer in MPU-60X0 assembled in the SensOne, includes a wide range of features:

- Digital-output triple-axis accelerometer with a programmable full-scale range of ±2g, ±4g, ±8g and ±16g
- Integrated 16-bit ADCs enable simultaneous sampling of accelerometers while requiring no external multiplexer
- Accelerometer normal operating current: 500µA
- Low power accelerometer mode current: 10µA at 1.25Hz, 20µA at 5Hz, 60µA at 20Hz, 110µA at 40Hz
- Orientation detection and signalling
- Tap detection
- User-programmable interrupts
- High-G interrupt
- User self-test

1.1.42 Inertial Measurement Unit accelerometer sensor

The third type of sensor that can be used to acquire raw accelerometric data is the inertial measurement unit (IMU), which is an electronic device that measures and reports a body's specific force, angular rate and sometimes the magnetic field surrounding the body, using a combination of accelerometers and gyroscopes, sometimes also magnetometers.
CRF has provided vehicles equipped with “PCAN-GPS”, a programmable sensor module for position and orientation determination. It has a satellite receiver, a magnetic field sensor, an accelerometer and a gyroscope. The sampled data can be transmitted on a CAN bus and logged on the internal memory card.

The BMC050 is a fully-compensated electronic compass including a triaxial geomagnetic sensor and a triaxial acceleration sensor (6 degrees of freedom) that deliver excellent performance in very small size. The BMC050 allows for determining precise tilt-compensated geomagnetic heading information and for providing accurate acceleration sensor data.

Table 7 - Integrated electronic compass

Integrated electronic compass BCMC050 (PCAN-GPS)	
Stand-alone operation	supported
Resolution	10 bits
Programmable f-range	±2g / ±4g / ±8g / ±16g"
Zero-g offset	±80 mg
Sensitivity tolerance	±4 %
Accelerometer interrupts	- Data-ready (e. g. for processor synchronization) - Any-motion (slope) detection (e. g. for wake-up) - Tap sensing (e. g. for tap-sensitive UI control) - Orientation change recognition (e. g. for portrait/

	- landscape & face-up/face-down switching) - Flat detection (e. g. for position sensitive switching) - Low-g / high-g detection (e. g. for shock and free-fall detection)

3.6 UWB localization

1.1.43 Dutch pilot site

Ultra-Wideband (UWB) modules have become popular for localization purposes in recent years. The modules operate on a frequency range between 3244MHz and 6999MHz. The modules calculate the distance to each other based on time of flight. Using several of these modules makes it possible to track the position of another module. This is also possible, if one module is static and the other one is moved over time. By collecting this data, one can estimate the relative position between the two modules. UWB is quite robust to multipath and no-line-of-sight effects, which makes them interesting to use in structured and unstructured outdoor environment. Since building up an infrastructure with UWB is rather straightforward, they are well suited for pose and map initialization. In AUTOPILOT, several static UWB-modules will be mounted on the side of the parking lot. AUWB-module that is mounted on the front side of the micro aerial vehicle (MAV) is measuring the time of flight (ToF) to the static UWB-modules and using multilateration to calculate its own position. The obtained measurement is then fused with other sensors on the MAV.

1.1.44 Finnish pilot site

The Finnish pilot site uses UWB technology under development by HERE for improving location accuracy of the automated vehicle. A network of UWB beacons is installed along the edges of the parking area. Two UWB receivers are installed at the vehicle roof, assuring accurate vehicle heading. The vehicle calculates its position based on the signals received from the fixed UWB beacons. This position is made available through the in-vehicle network to the other vehicle applications.

3.7 Crowd detector device with Wi-Fi and GPS sensors

1.1.45 Dutch pilot site

The task of the wireless sniffer is to receive Wi-Fi probes or Bluetooth messages from the surrounding environment. For AUTOPILOT, the surrounding environment refers to the vicinity of the autonomous vehicle since the wireless sniffers will be placed on the autonomous vehicles.
The wireless sniffer device is developed using Raspberry Pi (e.g. Raspberry Pi 3) devices with internal software to collect Wi-Fi probes and GPS locations with GPS sensors.
The figure simply showcases the usage of wireless sniffer for the purpose of VRU detection. The purpose of this device is to receive signals from mobile devices of pedestrians or cyclists. Smartphones or tablet computers can be examples of such mobile devices. The Wi-Fi mode should be enabled for the mobile devices.
Another example could be pedestrians or cyclists carrying Bluetooth Low Energy (BLE) beacons which broadcast Bluetooth messages. An alternative for Bluetooth broadcasting is that some smartphone applications provide beacon feature, such that the smartphone can act as a beacon broadcasting Bluetooth messages.

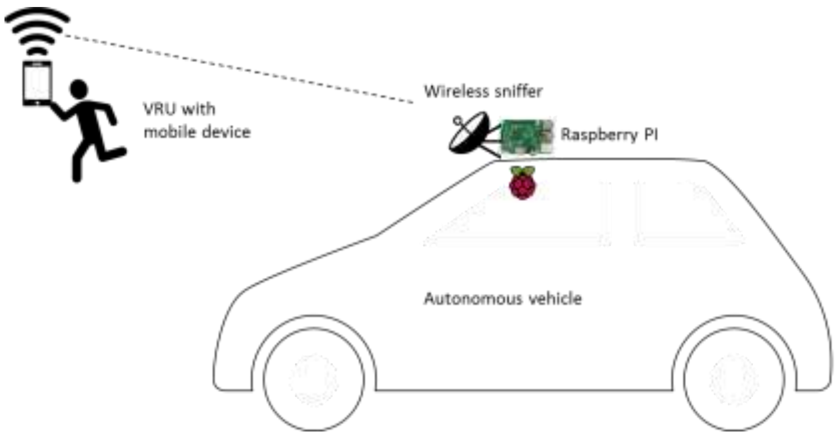

Figure 22 – Wireless sniffer receiving wireless signals from VRU.

The data received by the wireless sniffer can be stored initially in the buffer storage of the Raspberry Pi. Then, the data is forwarded to the IoT platform and Crowd Estimation and Mobility Analytics (CEMA) server-side service through 3G, 4G, or other available communication technologies. The data can be sent as chunks of data or the transfer can happen for every detected wireless package.

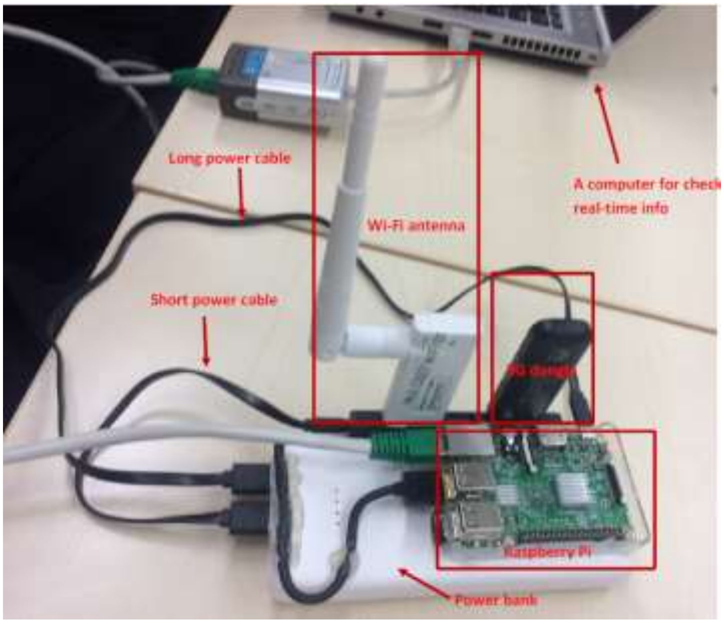

Figure 23 – An overview of the NEC crowd detector device.

For the Urban driving / Car rebalancing use case, the wireless sniffer device is used for the purpose of VRU-crowd detection. For instance, if based on the crowdedness levels of a university campus, the car can decide to take or not to take a certain route. The number of people is estimated based on the unique MAC addresses that are received by the wireless device for a certain period of time (e.g. 5 seconds). Figure 23shows the setup for Wi-Fi sniffer in the lab environment, which is named as the NEC crowd detector device. The device has a power cable (or a power bank if necessary) and a Raspberry Pi that runs the code for receiving the Wi-Fi probes. These include MAC addresses as well as Received signal strength indication (RSSI) levels. Moreover, the device has a GPS sensor to detect the location of the vehicle during the measurement of the crowd detector device. The received probe is anonymized (with hashing and salting mechanisms) on the Raspberry Pi device. The anonymized Wi-Fi and GPS data is then sent to the serve through the 3G dongle. In the setup in Figure 23, a computer receives the data and it has the server-side component of CEMA running.

3.8 BLE Beacons

1.1.46 French pilot site

BLE beacons are used on the French pilot for the vehicle to provide audio/video content when going past a point of interest. There will be 14 points of interest in total in front of which there will be up to 30 Bluetooth Low Energy beacons to identify them. The characteristics of the beacons are [26]:

- Size: 84x84x24mm
- Radio protocol: BLE
- Emission zone: up to 100m

Figure 24 - BLE beacon used for the Point of Interest notifications

The BLE Beacons will be installed as close as possible to the road, on street furniture (candelabra, poles, etc.), at 2 to 3 meters off the ground.

1.1.47 Dutch pilot site

Bluetooth Low Energy (BLE) beacons can be used for various purposes including parking location detection, accurate and autonomous parking as well as pedestrian or cyclist detection using the Bluetooth sniffers.
Beacon technology is based on broadcasting BLE profiles of the devices, which can be (in most cases) configured either manually or dynamically. Beacons enable broadcasting information at a certain frequency and with a certain signal strength, which could also be configured as the context of the information itself. The broadcasted messages can be received by other Bluetooth device. One can use the BLE beacons to realize the physical web or internet of things such as objects that are normally not connected. For example, a small beacon device can be attached to an everyday object, which normally does not have any communication capability, and then the object will start sending information. Smart displays can be equipped with beacons and they can be configured, and certain URLs can be broadcasted to show additional information about the content that wants to be shown. These scenarios are considered for tourism-related use cases.

Figure 25 – Beacon devices: nRF51822 and Raspberry Pi Zero W.

Beacons can come with a battery that has a certain lifetime. In this case, the lifetime of the beacon functionality depends on the size of the battery. On the other hand, some beacons can harvest energy using small solar panels. In the case of existence of electricity and plugs to connect, the

beacons can be directly plugged. For autonomous vehicles or for the road infrastructure, this could be the case. Therefore, two example devices are considered for realization of the BLE beacon technology: nRF51822 and Raspberry Pi Zero W. The computing power of these devices would also enable configuring the information broadcasted from the beacon as well as other properties of the beacon dynamically. The above figure shows these two devices. They are very small in size and they can easily be attached to different objects. Raspberry Pi Zero W connects directly to the IoT Platform. The information provided by the beacon can be directly managed by FIWARE. It can also act as a BLE sniffer. Thus, it can provide the platform with information about nearby beacons. nRF51822 is smaller and its battery lasts longer. It needs a second device (e.g. Raspberry Pi) that connects to it through Bluetooth and that acts as a relay to/from the IoT platform.
NEC developed an iPhone application that is also calibrated to understand the distance of the beacon accurately. The beacon devices are also registered to the FIWARE-based IoT platform so that they can be discovered through entity lookups.

3.9 NB-IoT sensors

1.1.48 Italian pilot site

Narrow Band IoT (NB-IoT, also known as LTE Cat NB1), is a new radio technology that enables Low Power Wide Area Network (LPWAN) devices to be connected using LTE cellular network, as shown in Figure 26.

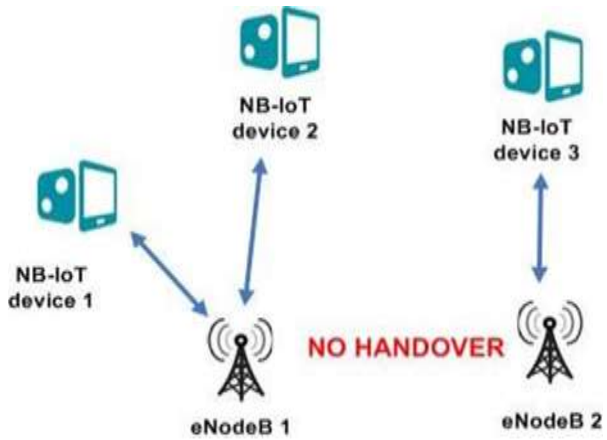

Figure 26 – NB-IoT scenario

The NB-IoT standard was introduced at Release 13 of the 3GPP specification in June 2016.
NB-IoT devices must accomplish the following constraints:

- Low power consumption: the expected battery life is 10 years
- Low throughput: maximum values are 144 kbps in Downlink and 200 kbps in Uplink
- Low cost
- Relaxed latency: up to 10 seconds
- Scalability: each base station should be able to handle 50.000 NB-IoT devices

Figure 27 shows the three types of frequency spectrums supported by NB-IoT:

- Stand-alone: channels derived from GSM reframing are used;
- Guard-Band: guard bands of LTE spectrum are used;
- In-band: Physical Resource Blocks (PRBs) inside the LTE spectrum are used.

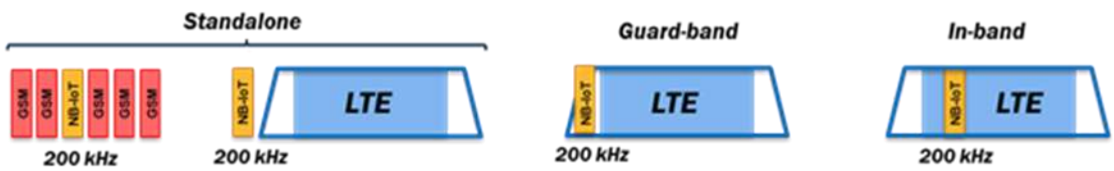

Figure 27 – NB-IoT channels

As reported in Figure 27, the NB-IoT standard does not support the handover procedure. Thus, Low-Power-WAN based on this technology could not be designed for in-vehicle applications.
Indeed, in the context of the Italian pilot site, it is planned to employ NB-IoT smart objects to collect data from water level sensor installed at roadside and forward it to the OneM2M platform, provided that the base stations are updated to support this technology. The sensing element of the NB-IoT puddle detector is shown in Figure 28.

Figure 28 – Water level sensor model connected to the NB-IoT puddle detector

Its main technical features are listed in Table 8.

Table 8 – Water detector main features

Parameter	Value
Working voltage	3.3/5V
Output voltage range	0 ÷ 2.3V
Current	< 20mA
Size	65mm × 20mm × 8mm
Detection area	40mm × 16mm
Weight	3g

The output voltage is an analogue value proportional to the measured level of water. Figure 29 shows how this value is processed when it is captured by the smart object.

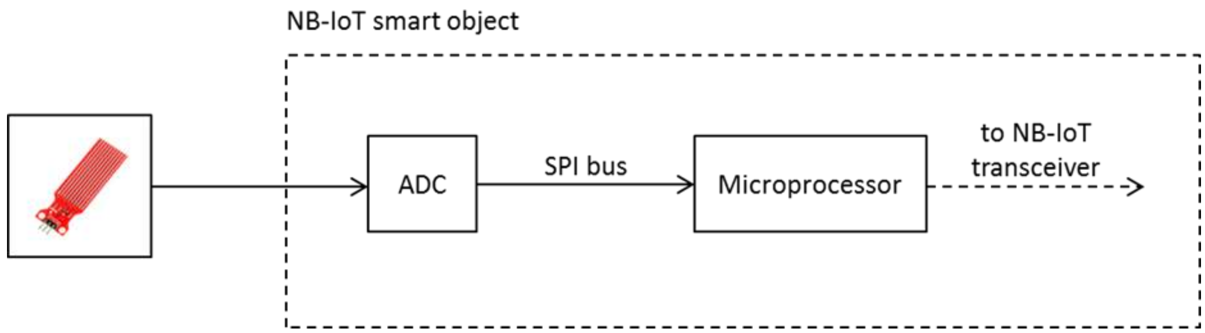

Figure 29 – Processing of the measured level of water

The output voltage produced by the sensor is converted to a digital signal by means of a 24-bit Analog to Digital Converter. Then, the digital value is sent to the smart object microprocessor via an

SPI bus.

According to the received digital value, a sensor driver installed in the microprocessor (programmed during a previous configuration phase) detects if the water level on the roadway surface is higher than a deterministic threshold or not. In the positive case, the driver associates to the digital value the Alarm Level 1, otherwise the Alarm Level 0. When a state transition occurs (from Alarm Level 0 to Alarm Level 1 or vice versa), the microprocessor sends to the NB-IoT transceiver (via AT commands) a message containing the alarm level type.

A simple firmware developed by CNIT handles the communication between the microprocessor and the NB-IoT transceiver: it converts the water level measurements of the sensor in two alarm levels (Alarm Level 0 or Alarm Level 1), according to a deterministic threshold. Then a payload containing the alarm level, the position and the time is formed and sent to the oneM2M platform using the AT commands of the modem and UDP or CoAP protocols.

The different components of the NB-IoT puddle detector developed by CNIT are shown in Figure 30.

Figure 30 – NB-IoT smart object developed by CNIT. 1) Quectel BC95-B20 transceiver; 2) Evaluation board; 3) Management board; 4) Battery; 5) Power input; 6) On/Off button; 7) Sensor input; 8) Antenna

The management board operation is based on the low power MCU ARM® Cortex®-M3. The NB-IoT module is constituted by a Quectel BC95-B20 transceiver placed on an evaluation board. The BC95-B20 key features are listed in Table 9.

Table 9 – Quectel BC95-B20 key features

Feature	Implementation
Power Supply	3.1 V ~ 4.2 V
Transmitting Power	23 dBm ± 2 dB
Temperature Range	-35 °C + 75 °C
Size	19.9 ± 0.15 × 23.6 ± 0.15 × 2.2 ± 0.2 mm
Weight	~ 1.6 g
UART Interfaces	Main port: used for AT command communication and data transmission. It only supports 9600bps baud rate Debug port: used for debugging. It only supports 921600bps baud rate

Operating Frequencies	Receive: 791~821 MHz Transmit: 832~862 MHz
Receive Sensitivity (RSRP)	-135 dBm

3.10 LoRaWAN

1.1.49 French pilot site

The parking spots of the car sharing stations in Versailles will be equipped with parking detectors so that the intelligent fleet management system gets the information on how many vehicles are available on each car sharing station. These detectors are installed in the ground and work through LoRaWAN technology. Other characteristics are:

- Directive antenna yagi 2.4 GHz
- 868 MHz antenna
- Magnetic detection
- LoRaWAN, RFID and Bluetooth connectivity

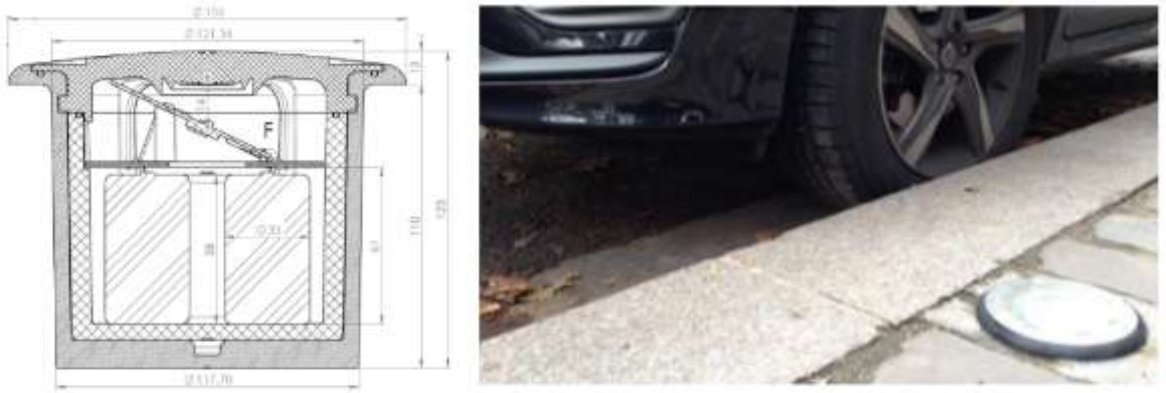

Figure 31 - Parking detector (ONESITU) used on car sharing stations

4. Use cases and functions for IoT devices and the pilot sites

Chapter 4 describe the functions of IoT devices and the use cases in which they are implemented in the different pilot sites. This includes connectivity with IoT devices, connectivity between vehicles, infrastructure and other sensors to enhance autonomous driving capabilities and technology that allows vehicles to monitor the state and availability of different services. As for the in-vehicle functions, three main groups of sensor systems such as camera-, radar- and lidar-based systems, together with ultrasonic sensors are used for autonomous driving.

Chapter 4 presents how other type of sensors/actuators and IoT devices in the different use cases are used to enhance the autonomous driving capabilities.

As the name suggests, autonomous driving allows cars to operate with varying levels of user intervention.

An overview of the pilot sites and their respective use cases are given in Table 10. Each use case includes a number of IoT sensor and actuator devices. These use cases and their functions are described below according to pilot sites.

Table 10 – Pilot sites and use cases

Use cases	Versailles (FR)	Livorno (IT)	Brainport (NL)	Vigo (ES)	Tampere (FI)
Automated valet parking			X	X	X
Highway pilot		X	X		
Platooning	X		X		
Urban driving	X	X	X	X	X
Car sharing	X		X		

In addition, the expected increase in the AD-level capabilities for each pilot site and their use cases are indicated and explained according to the SAE standard J3016. The automation levels are given in Table 11 [25].

Table 11 - SAE automation levels [25]

0 No Automation	1 Driver assistance	2 Partial Automation	3 Conditional Automation	4 High Automation	5 Full Automation
Human driver monitors the driving environment			Automated driving system monitors the driving environment		
Zero autonomy, the driver performs all driving tasks.	Vehicle is controlled by the driver, but some driving assist features may be included in the vehicle design.	Vehicle has combined automated functions, like acceleration and steering, but the driver must remain engaged with the driving task and	Driver is a necessity but is not required to monitor the environment. The driver must be ready to take control of the vehicle at all times with notice.	The vehicle is capable of performing all driving functions under certain conditions. The driver may have the option to control the	The vehicle is capable of performing all driving functions under all conditions. The driver may have the option to control the

		monitor the environment at all times.		vehicle.	vehicle.

4.1 French pilot site

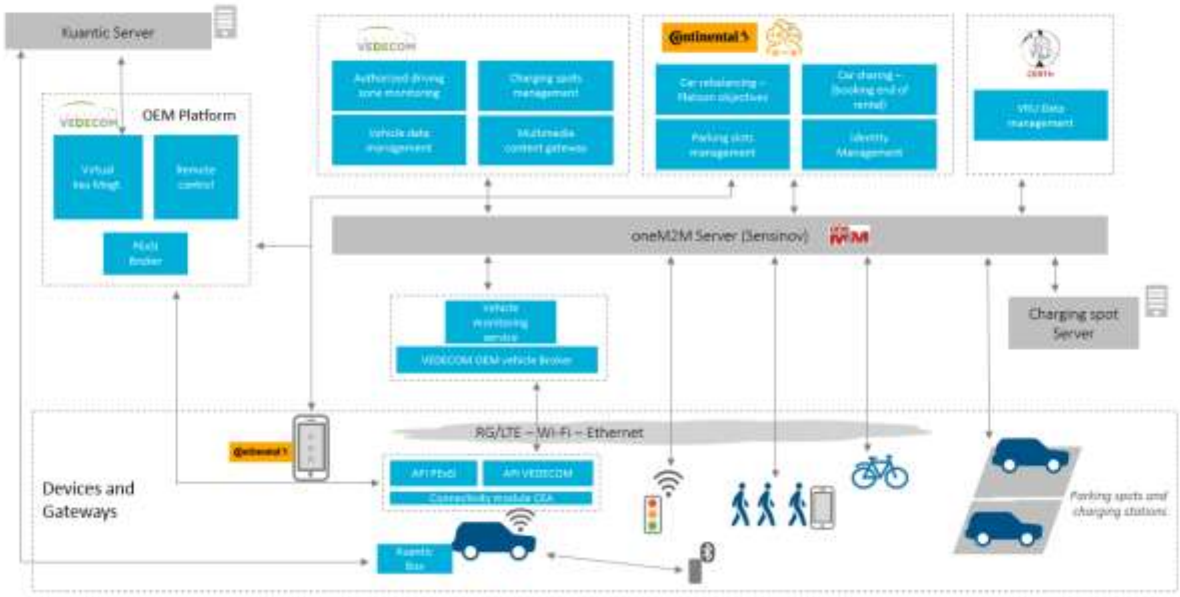

Figure 32 - Versailles pilot site architecture overview

The French pilot site is situated in the city centre of Versailles. The goal is to provide a mobility service dedicated to visitors of the city and the castle's gardens. Three car sharing stations will be deployed (one in front of the town hall and two at opposite entrances of the castle's gardens). A fleet of five connected and autonomous vehicles (Renault Twizy) will be developed for the AUTOPILOT use cases. The users will be able to download a smartphone application in order to reserve one of the vehicles and go on an urban trip to discover Versailles and its historic monuments, churches, walk paths and more. Three main use cases are being developed:

- Car sharing for touristic applications with three stations equipped with electric charging points;
- Urban driving: connected and automated driving in the city centre and the castle's gardens with point of interest notifications (audio/video) and VRU detection (collaborative perception);
- Platooning for automatic fleet rebalancing between the three stations.

1.1.50 Car sharing use case functions (Versailles)

The car sharing use case is about offering a car sharing service for tourists visiting Versailles and the Castle's gardens. It also supports the two other use cases (urban driving, platooning) deployed on the French pilot site.
Within this use case, the added value of IoT to the quality of service offered to the users is going to be analysed. Indeed, the use of IoT is expected to assist responding to the demand of having a sufficient number of vehicles in different stations. Through parking sensors and charging stations, the IoT devices are enabling a real-time parking management thanks to the common IoT platform. The devices are pushing automatically their status allowing a faster fleet management. Regarding vehicles, considered as made of several IoT devices, they are pushing on the common IoT platform their own status (level of battery, localization) and allow to the car sharing cloud service to deliver a

better booking Application.
The objective is on one hand to increase the quality of service for the users and on the other hand, to reduce the exploitation costs. Actually, all partners pushing their IoT devices data allow to all potential smart cities use cases to be more efficient. Also, the car sharing service and the data produced are used as input for the platooning use case, through car rebalancing service.
The Car sharing use case aims at providing a service of vehicle proposal and availability to users, directly linked to IoT integration. However, this service does not allow to evaluate directly performance of AD.

1.1.51 Platooning use case functions (Versailles)

The Platooning Use Case is part of the car rebalancing business case. It is closely linked to the fleet management system that indicates which vehicles have to be transferred from one station to another.
The added value of the Internet of Things in the platooning use case is illustrated in the following aspects:

- Mission planning:
  - Choose the leading vehicle and its start/end stations according to data collected via IoT objects (e.g. the position of the operator, the charging level of the vehicle, etc.).
  - Choose the follower vehicles, the start/end station and their order in the platoon according to data collected via IoT sensors in each car and in the parking spots.
- Traffic Light Assist:
  - Suggest a reference speed to the operator in order to minimize the waiting time (red light) at each intersection that counts with a traffic light along the entire itinerary.

1.1.52 Urban driving use case functions (Versailles)

The scope of the Urban Driving Use Case is to show connected and automated driving in an urban environment in legacy traffic. All roads, except of one, are located in the urban environment of the city of Versailles. The other one is located in the castle’s gardens and is only shared with vulnerable road users (pedestrians, cyclists). Connected driving is possible on all roads of this itinerary; autonomous driving only in the castle’s garden where collaborative perception will be tested.

The users having rented a vehicle at one of the car sharing stations are going to receive audio point of interest (PoI) notifications when driving through the city centre in manual mode, and audio plus video PoIs notifications when in AD mode.

Some of the vulnerable road users (VRU) will be equipped with smart devices such as smart watches, smart glasses and/or smartphones. These devices are considered as ‘IoT objects’ communicating via the IoT platform. Some VRUs will also ride a bicycle equipped with an on-board unit for direct communication with the AD vehicle. The latter will receive CAMs and DENMs from the bicycles, and the communication system will provide the information to the autonomous driving system that is developed by VEDECOM.

4.2 Dutch pilot site

The Brainport pilot site concerns the region of Helmond-Eindhoven in the Netherlands. The region includes three campuses (Eindhoven University, Automotive campus, and High-Tech campus) and Eindhoven airport. The main road between the cities of Eindhoven and Helmond is the A270 motorway, which is part of the DITCM (Dutch integrated test site cooperative mobility) test site. The DITCM test site is a purpose-built facility for the development, testing and validation of Intelligent Transport Systems (ITS) and cooperative driving technologies. It consists of both a motorway (A270 and N270) and urban environments. The test site is 8 km long, with 6 km of motorway. The Brainport pilot architecture (simplified) is depicted below.

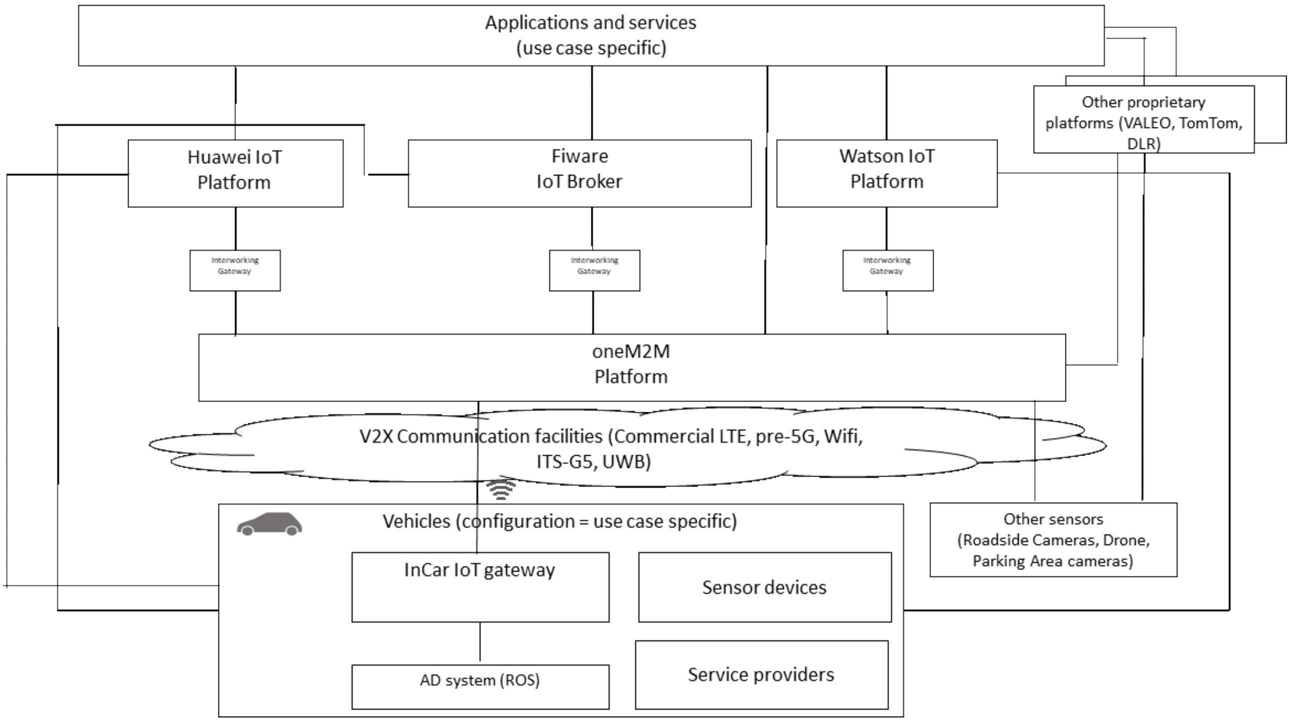

Figure 33 - The Brainport pilot architecture (simplified)

1.1.53 Automated valet parking use case functions (Brainport)

In the AUTOPILOT-project, the AVP use case will take place at the Helmond Automotive Campus. The use case story starts with the vehicle being manually driven to the drop-off point in front of the TNO building. After arriving there, the user activates the AVP function (e.g. by in-vehicle interface or smartphone app) and exits the vehicle. Services on the IoT platform determine an obstacle-free route to an available parking position based on information from IoT devices. The vehicle autonomously drives to the dedicated parking position. IoT devices involved in the use case are:

- Permanently installed cameras on the parking area that can detect free parking spots and obstacles;
- A micro aerial vehicle (MAV) that can provide information about free parking spots and obstacles, in particular for areas the cameras do not cover;
- IoT-enabled vehicles with own sensors.

The primary goal of the IoT usage is therefore to gain an improved environment model that can possibly increase efficiency and safety of the use case.

Figure 34 depicts an overview about the IoT architecture of the AVP use case as deployed in

Brainport. Two IoT platforms from Watson IBM and oneM2M are used by the AVP and the interoperability between the two platforms is realized through the interworking bidirectional connector that has been implemented for this purpose.

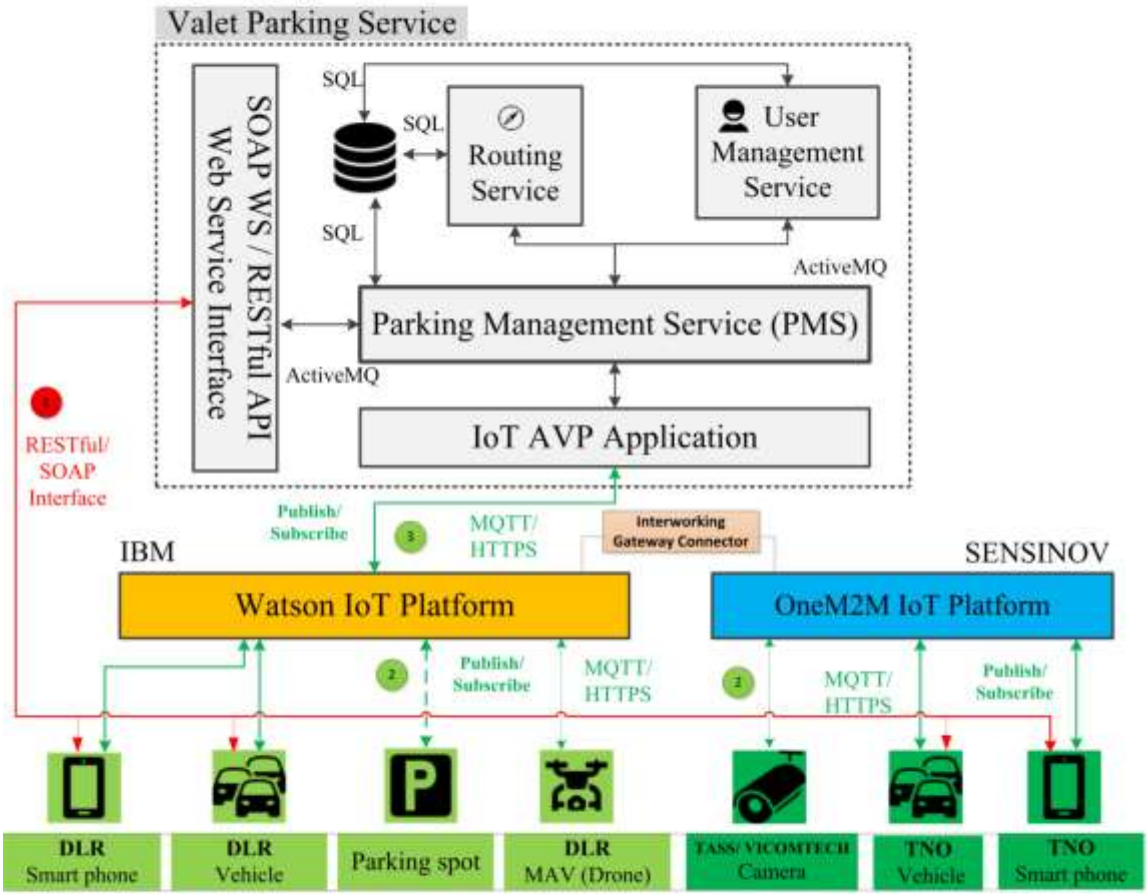

Figure 34 - Automated valet parking use case IoT architecture

1.1.54 Highway pilot use case functions (Brainport)

For the Detection component of the System, three sensors in the car are relied upon: a Lidar, a front camera and an IMU. An extra camera supports the use case for lane detection but is not directly involved in hazards detection. The Lidar data is processed by a specific algorithm developed by Vicomtech, focusing on speed bumps detection.

The front camera data is processed by a specific algorithm developed by Vicomtech, focusing on potholes. The IMU data is processed by a specific algorithm developed by Valeo, capable of detecting anomalies without specific classification. For the Information component of the System, one actuator is relied upon: the ACC control unit. Moreover, turning lights are controlled to support lane changes scenario.

The way all these are interconnected is illustrated in the following picture of the in-vehicle SW and IoT architecture.

It is worth noting that the raw data from sensors is indeed passed directly to the runtime environment where the real-time detection algorithms run. However, everything else is coordinated through an in-vehicle IoT platform (here a MQTT Broker) that ensures the coordination between the results from all other software modules.

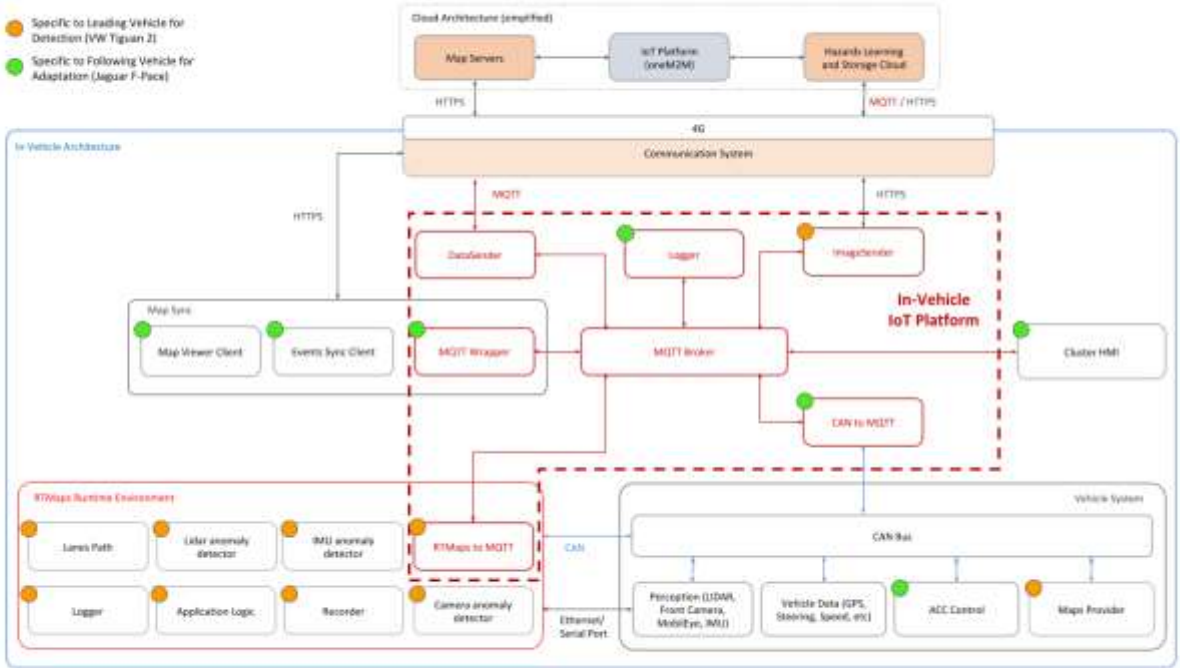

Figure 35 - Software and IoT architecture in Valeo prototypes

In addition to IoT devices within vehicles, the use case also takes advantage of a roadside camera that monitors the road for anomalies too (e.g. static objects like fallen cargo). The detection from this camera is passed though TASS onto the OneM2M IoT Platform directly.

1.1.55 Platooning use case functions (Brainport)

The main scope is to show how increased flexibility in platoon navigation and manoeuvring capabilities can be realized, and how it can benefit from the use of IoT technology. For instance, platoon forming is done under control of a Platoon service that provides route and speed advice, and recalculates estimated time of arrival and pick-up/drop off point on the basis of the actual positions and speeds of the vehicles. Additional achievement is to guide the platoon after successful formation. Guidance involves speed and lane advice to the lead vehicle, based on the traffic situation on the A270. For example, the Platoon service receives regulatory information from the road operator (max speed and lane access/ or closure) and also takes data from the IoT platform (oneM2M) concerning vehicle traffic conditions and traffic light status data. In order to minimize the probability of platoon break-up, the Platoon service provides a specific speed advice. After a break-up, it will support reformation. The use case also involves the use of the hard shoulder but at the time of writing this deliverable, this is considered quite challenging so an existing special purpose lane (e.g. bus lane) may be used instead. The platooning use case uses various communication channels (V2V and V2I). V2V concerns operational ACC while the bidirectional V2I channels are mainly used for tactical data exchange. Relevant IoT data are the road operator originated info, the actual Traffic State data (through A270 camera array), platoon state data and traffic Light data. Road Operator Logging takes place on the vehicle (vehicle state and control) and on the IoT platform.

The execution view of the systems and processes involved during the platoon formation stage gives some insight into the system architecture implemented for Platooning. The intended procedures in Figure 36 are:

1. Traveller steps into the car and starts the Car sharing app;
2. Traveller defines whether he/she wants to be leading or following in platoon;

3. Traveller defines the destination;
4. Car sharing app already knows about existing platoons and can match;
5. Car sharing app gives route to Watson IoT, which sends it too oneM2M;
6. Traveller presses the vehicle GUI to put the vehicle in platooning joining/leading mode;
7. Platoon service receives message from the vehicle that it wants to platoon;
8. Platoon service app receives message from the car sharing app that match has been made;
9. Platoon service app gives route(s) to the planner => fill platoon formation message with info from planner and send to vehicles;
10. Vehicle receives platoon formation message containing platoon ID and planner information.

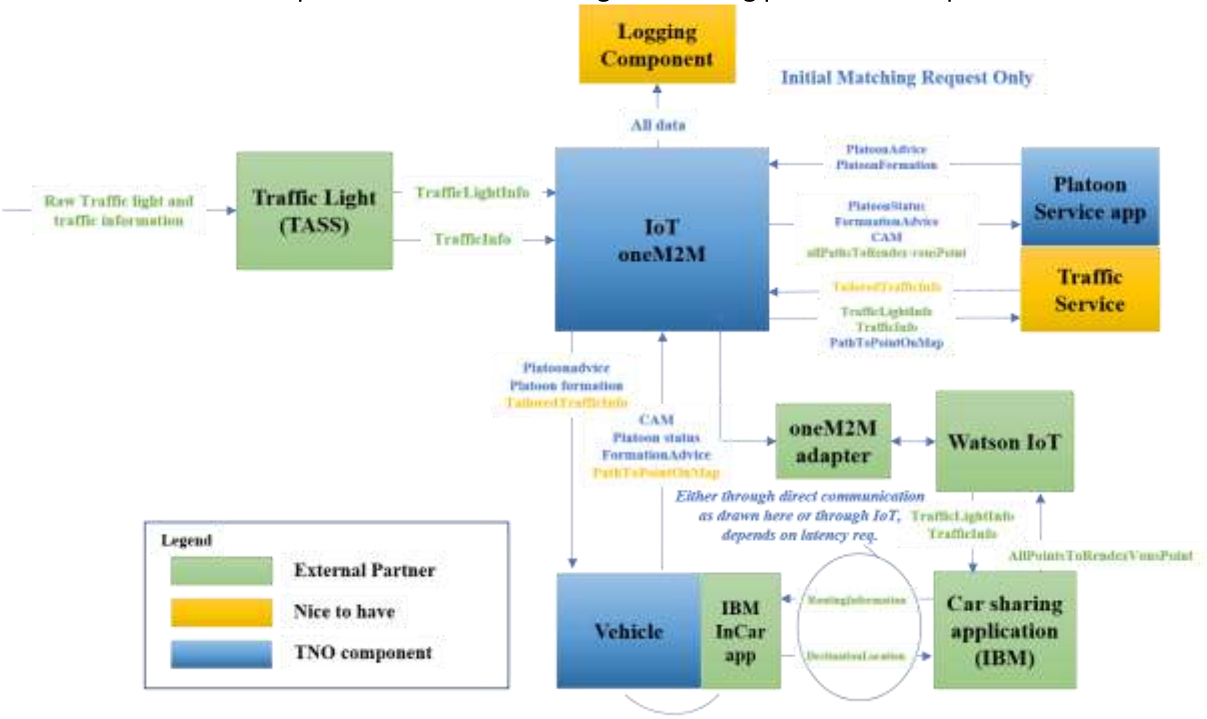

Figure 36 - Platooning use case execution view of Platoon formation

The Platoon service of TNO listens to the cloud-based Traffic Manager application provided by TASS, which delivers regulatory road information.

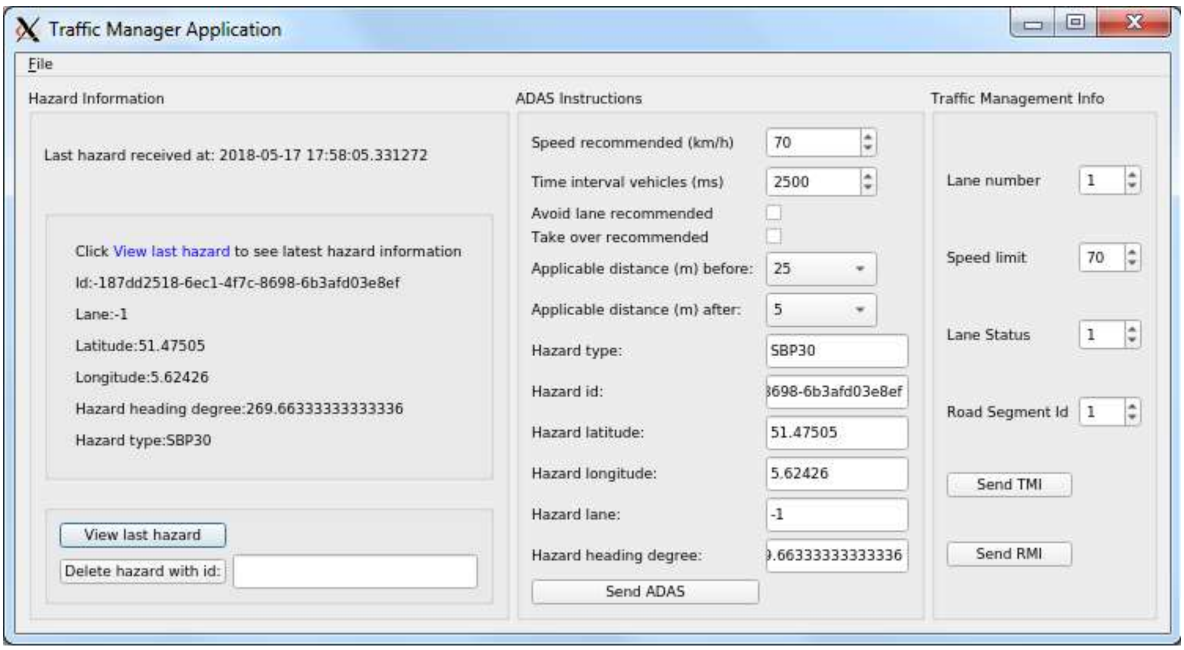

Figure 37 - GUI of the Traffic Manager application (TASS)

The traffic operator (person) can update the Traffic Management Info such as speed limits, emergency lane status, etc., using the GUI and publishes this information to a respective container in OneM2M. The operator can also publish road map information (usually static) wherever there is any change to the otherwise static map (defined by TASS and TNO jointly). The platooning vehicles subscribed to these containers in OneM2M get these updates and adapts their driving accordingly. Below a screen shot is depicted of the GUI of the Traffic Manager application.

1.1.56 Urban driving / Car rebalancing use case functions (Brainport)

The main scope is to show how automated driving with vulnerable road users (VRUs) detection can be realized for Urban driving / Car rebalancing.

For Brainport, 3 different modalities for detecting VRUs are implemented: Crowd Estimation and Mobility Analytics using Wi-Fi-based measurements; smartphones enabled with IoT connectivity (OneM2M platform integration) and ITS-G5 (CAM messages)-enabled Technolution FlowRadar devices (which can be carried by pedestrians or mounted on bicycles).

Crowd Estimation and Mobility Analytics (CEMA) system is used in the use case for the purpose of enhancing the VRU detection and taking actions based on the “crowdedness” of the environment surrounding the autonomous vehicle. While Wi-Fi-based measurements are not very accurate due to noises and differences of environments such as different obstacles and wireless interferences, the CEMA outputs can be given as a feedback to improve the world model. For instance, if crowdedness is estimated by the CEMA system, the autonomous vehicle can consider it while taking a decision for a certain route. Moreover, multiple cars can share crowdedness information with each other to have a more global view of crowdedness. The CEMA crowd detector device is based on usage of the **wireless sensors and GPS sensors** and their deployment to the autonomous vehicle**.** Wireless sniffer devices are responsible to collected Wi-Fi probes from their vicinities and forward this information to the server-side of the CEMA system.

For the use case also, a pre-defined number of the vulnerable road users (VRU) will be equipped with IoT enabled smartphones. These smartphones are considered as ‘IoT objects’ communicating via the IoT platform (OneM2M and/or HUAWEI OceanConnect platform).

Mainly for verification, validation & benchmarking, some VRUs will be equipped with a portable ITS-G5 enabled unit (FlowRadar provided by Technolution) for direct communication with the AD vehicle. The latter will receive CAM messages from the bicycles & vice versa and the communication system will provide the information to the autonomous driving system of the AD vehicle.

Considering SAE levels of AD, the vehicle will be self-driving. However, since this is a research vehicle it still requires a trained engineer to monitor the system in case of safety related issues. IoT adds in this case extra redundancy, but it is to be tested in the Pilot Tests in how far this can be redundant to the existing sensor set in the vehicle. In best case, the aim is to increase from AD level 3 to a maximum of level 4.

1.1.57 Car sharing use case functions (Brainport)

A car sharing service is intended as a service to enable different customers to make use of a fleet of cars (either self-driving or not) which is shared amongst them. Car sharing can be interpreted as a service that finds the closest available car and assigns it to a single customer or drive the closest available car to the requesting customer. Car sharing can also be intended as ride sharing, when multiple customers that possibly have different origins and destinations share a part of the ride on a common car (either self-driving or by driving it themselves). Finally, car sharing services can also be thought of as services that allow customers to specify pick-up and drop-off time-windows to increase flexibility and planning.

Figure 38 shows the target architecture for the car sharing use case. The focus here is on the interaction between the various car sharing actors and components and the Open IoT platform common services as a whole, represented as one box.

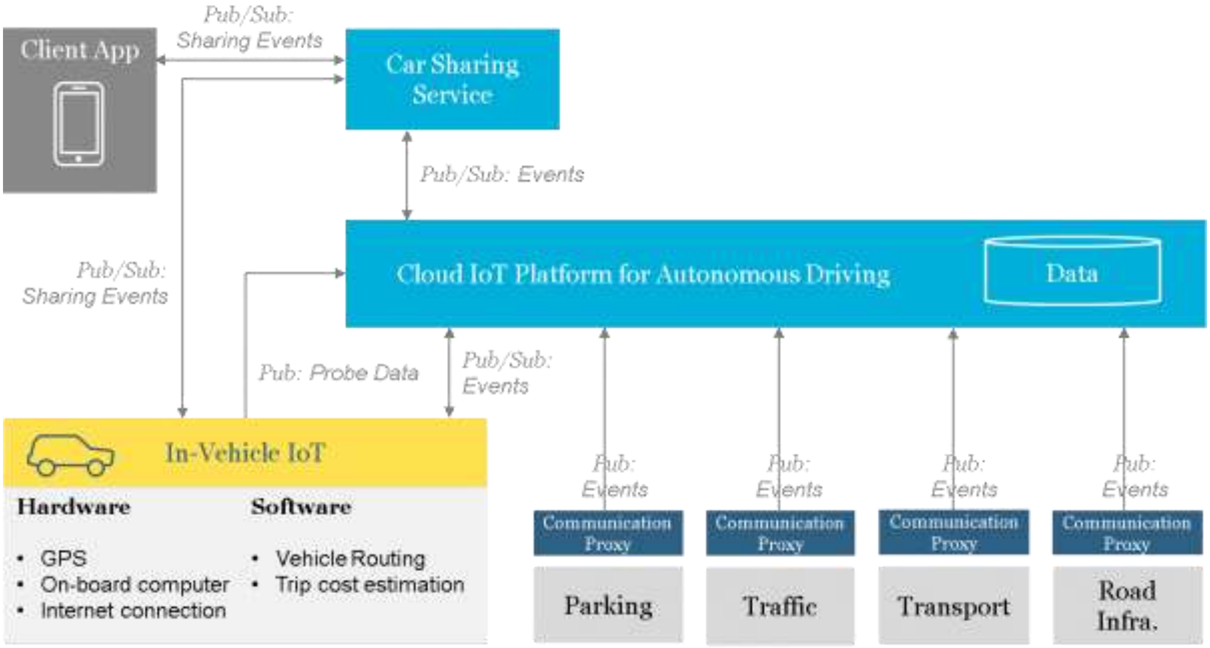

Figure 38 – Car Sharing Use Case Architecture

Users should book cars and manage (modify, cancel, etc.) their bookings using the central car sharing service through a mobile or desktop application, referred to as the client app.

The proposed architecture requires that shared vehicles should be equipped with the necessary hardware and software to: (1) communicate their probe data (GPS location, speed, etc.) to the open IoT platform common services and the car sharing service, and (2) compute optimum routes and their costs (distance, energy consumption, etc.) given an assigned destination. These may be fully implemented inside the vehicle itself or may be delegated to external web services.

IoT-enabled devices and vehicles of the IoT ecosystem should publish relevant events (traffic, accidents, weather, parking spot availability, etc.) on the open IoT platform. In order for the car sharing service and shared cars to be notified about events that may affect their planned trips, they should subscribe to the open IoT platform for relevant events. The open IoT platform should be responsible for collecting data from the various IoT devices, storing them and communicating the relevant pieces of data (events) to subscribers.

4.3 Italian pilot site

The Italian pilot site is a testing infrastructure encompassing the Florence-Livorno freeway together with road access to the Livorno sea port settlement. The testbed consists of three zones: The Livorno-Florence freeway, The Traffic control centre (TCC) located in Empoli, and the port landside just in front of the cruise terminal. The vehicles which will be used in the test site are FCA Jeep Renegade with different functions and roles: two vehicles by CRF with automated driving functions and five service vans by CRF and AVR with advanced V2X communication capabilities. Both Highway pilot and urban driving use cases are performed. The Pilot Site architecture is shown in Figure 39.

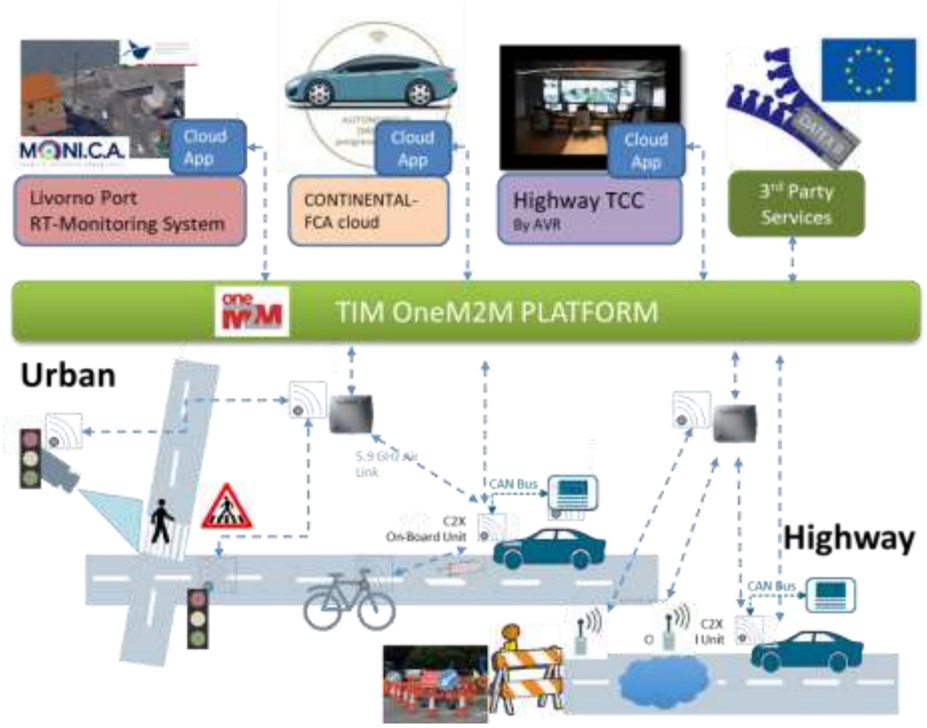

Figure 39 – Livorno pilot site architecture

It can be observed by looking at Figure 39 that adding IoT devices to the AD context extends the attack surface related to cybersecurity threats. AUTOPILOT is not going to develop novel security devices but will focus on developing and integrating devices and protocols that employ established and well known IoT and embedded systems solutions to security problems. Deliverable D1.9 (Initial specification of Security and Privacy for IoT-enhanced AD) provides a security risk analysis that is intended to drive the development of the AUTOPILOT features while providing the expected security and privacy that are crucial for AD. Because of the limited effort scheduled by the AUTOPILOT project on the implementation of security features, it is possible that not all of the D1.9 security requirements will be met at project completion. The starting point, as of the date of the internal diffusion of this document, sees most of the IoT devices and their connections still not secured and the compliance with many security requirements is still under evaluation. For example, the ITS-G5 messages sent by the vehicles and the infrastructure elements are not already using the security functionalities described by the ETSI standards. Using the D1.9 risk analysis, the next stages of the project will focus on mitigating the most dangerous threats while taking into account the effort needed to remedy the situation in a balanced cost-benefit way.

1.1.58 Highway pilot use case functions (Livorno-Florence)

The scope of these tests involves cars with IoT enhanced AD functions, driving in a “smart” highway. The cars are Jeep Renegades with on board equipment, (the so-called IoT open vehicular platform) enabling IoT triggered AD functions, namely: speed adaptation, lane change, lane keeping. Some cars also have special sensors, such as the IoT-based pothole detector.

The “smart” highway is a freeway where a pervasive IoT ICT system is deployed based on a network of roadside sensors or other sources capable of collecting information and making it available to cloud-based applications. In the use cases, connected cars and the Traffic Control Center also have an important role. For safety reason, the connected cars precede and follow the AD car driving in convoy.

The goal is to show how the combined use of IoT and C-ITS can mitigate the risk of accident for an AD car, when at a certain point, the road becomes dangerous because of two kinds of hazardous events: 1) wet road, 2) road works. In the following, the functions of the different IoT devices are described.

1. **Hazard on the roadway (puddle)**

- Puddle IoT sensors:
  - In the Italian PS, two kinds of such sensors are deployed, using different communication technologies: 6LowPAN and NB-IoT. They continuously monitor the Highway in critical locations sending two kinds of signals: a low frequency heart beat and a high frequency alert triggered by the rising of the water level;
  - The 6loWPAN puddle sensors send the messages to the Road Side ITS-Station by means of CoAP;
  - The NB-IoT puddle sensor sends the message straight to the oneM2M platform using the LTE cellular network and REST protocols.

- Road Side ITS-Station:
  - Road Side ITS-Station is a programmable gateway with multi-access technologies (notably 6LowPAN, ETSI ITS G5, LTE, Ethernet, etc.). It is a RSU, compliant with ISO/TC204 WG16 standards, able to exchange information over different networks, using different protocols, including the IoT ones;
  - The RSU always listens to the 6loWPAN sensors and sends the measurement to the OneM2M IoT platform of the PS with a certain frequency;
  - When a hazard occurs, the RSU broadcasts a DENM with the lowest quality level of the information (i.e. not yet validated by the TCC), toward both the approaching vehicles via the ITS-G5 network, and the oneM2M platform via LTE cellular network.
  - Furthermore, the RSU publishes on the oneM2M platform the CAMs collected from the vehicles in the ITS-G5 communication range.

- The Traffic Control Centre:
  - The TCC implements a DATEX II node that is allowed to supply information from the whole highway network. The TCC is also responsible for managing ITS on the oneM2M platform of the Italian PS. Two kinds of services are provided leveraging the subscription to the oneM2M platform: hazard validation, DENM forwarding. It also publishes to the oneM2M platform the relevant traffic information from the DATEX II node, to be consumed by the Highway infotainment service (Fi-PI-LI App);
  - When a hazard (flooding on the road) occurs, the TCC is notified by the subscription to the oneM2M platform. After assessing the severity of the danger, it validates the hazard and broadcasts a DENM with the highest quality level of the information (i.e. validated by the TCC) to the RSUs along the Highway, using the cabled LAN;
  - The TCC subscribes to the CAMs of the vehicles published by the RSUs on the oneM2M platform. The information is combined with the Bluetooth and Wi-Fi transit data loggers in order to perform the travel time analysis and live overview on the TCC video wall;
  - The TCC subscribes to the AD car's sensor data on the oneM2M platform in order to provide ITS services to the users of the highway.

- The Autonomous Driving car:
  - The AD car broadcasts CAMs over the IEEE 802.11 OCB (ETSI ITS-G5) network; at the same time the AD car publishes data from its sensors to the oneM2M platform;
  - The AD car is approaching the hazard on the road: the in-vehicle application (Connected-eHorizon) subscribes to the alert from the oneM2M platform;
  - The in-vehicle IoT platform combines the information obtained by the CeH

(Connected-eHorizon) with that obtained by DENM via the IEEE 802.11 OCB (ETSI ITS-G5) network and then feeds the appropriate autonomous functions that perform either the necessary adaptation of the driving style in a “smooth” way, if sufficiently in advance;
    - On the other hand, if the vehicle is close to the hazard and for some reason (i.e. the warning from the IoT services wasn’t received, or the warning was received just by the safety channels of ITS-G5 (DSRC), an emergency braking is needed, this event is registered by the in-vehicle application and sent to the OneM2M IoT platform of the PS;
    - At the same time the FCA cloud monitors the performance of the vehicle, checks that the in-vehicle application feeds the appropriate autonomous functions, sends notification/warning to the in-vehicle HMI.

- The “connected” cars:
    - The connected cars lead and follow the AD car; they continuously broadcast CAMs over the ETSI ITS-G5 (IEEE 802.11 OCB) network; at the same time, they publish its sensor’s data the oneM2M platform;
    - The connected cars are approaching the hazard on the road; the in-vehicle IoT platform receives the information from both the RSU along the track and the OneM2M IoT Platform;
    - The in-vehicle application pre-alerts the driver about the hazard using the information obtained by the OneM2M IoT Platform of the PS and by the DENM.

An overview of the demonstration storyboard is shown in Figure 40: IoT sensors placed along the highway monitor continuously the presence of puddles and if a warning condition has been detected, send an alert to the Road Side Unit (RSU) that broadcasts this information to vehicles (DENM) and to the Traffic Control Centre (TCC). It validates the alert, forwards the DENM message to farther away RSUs and feeds the IoT OneM2M cloud platform with alert related data.
The information on the presence of puddles generates a temporary update of the speed limit in the interested area, which is transmitted from the CONTI/FCA Cloud to the Connected e-Horizon (CeH) installed inside the FCA prototypes. The in-vehicle application feeds the appropriate autonomous functions that perform a smooth speed adaptation (**IoT-enabled speed adaptation for AD car**) in combination with information obtained from DENM. In consequence, IoT technology assists the rising of the automation level from 3 to 4.

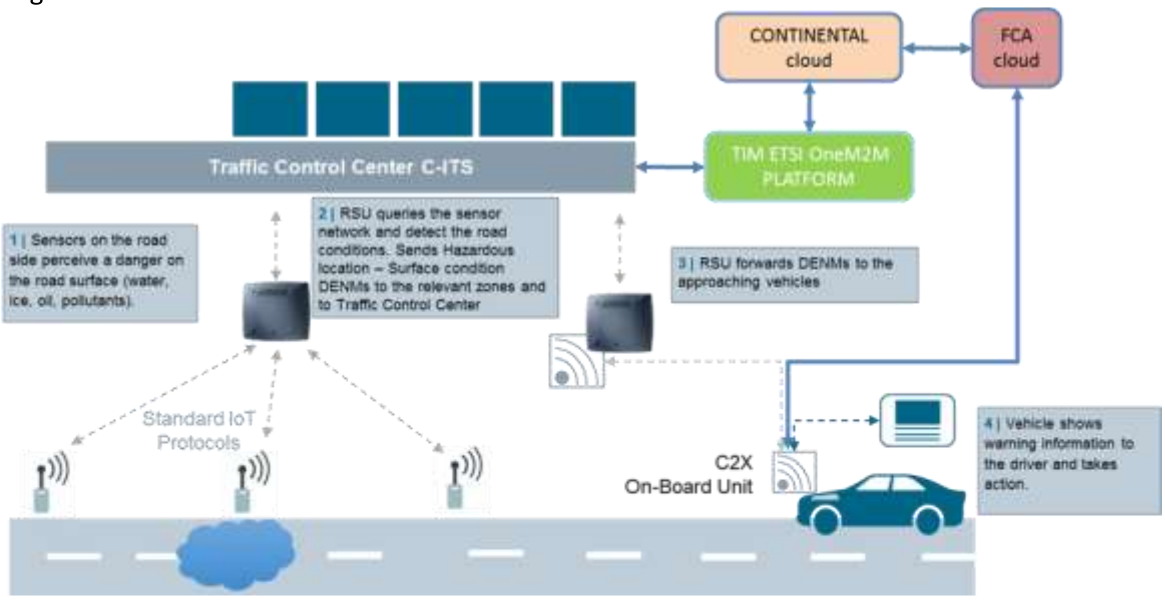

Figure 40 – Hazard on the roadway (puddle) execution view

2. Roadworks warning by TCC

A roadworks event is planned by traffic/road operator and a temporary speed limit is associated with the event. Two IoT-assisted AD manoeuvres are expected:

- The AD vehicle has to reduce its speed approaching the roadworks area, travel at the temporary speed limitation and increase again the speed at the end of the roadwork area;
- The AD vehicle has to stay on the current lane without any human steering action. Moreover, in presence of a lane closed due to roadworks, it has to perform a lane change and avoid the obstacle.

An overview of the demonstration storyboard is shown in Figure 41

- A sensor node is attached to the road works trailer and announces the presence of roadway works to an RSU;
- Then the RSU triggers DENM messages, broadcasting information about available lanes, speed limits, geometry, alternative routes, etc.;
- The TCC broadcasts the DENM messages to farther away RSUs. At the same time, the TCC feeds the OneM2M platform with road works-related data;
- Then the information is consumed by the Connected eHorizon (CeH) application from CONTINENTAL and transmitted to FCA cloud as a modified dynamic speed limit that considers the generated dynamic event;
- FCA cloud immediately notifies to enabled vehicles the updated information for CeH devices installed on prototypes, and the in-vehicle application feeds the appropriate autonomous functions that perform the necessary adaptation of the driving style in combination with information obtained from DENM. A notification/warning can be generated through the in-vehicle HMI;

As expected benefits, IoT can provide to the AD car information in advance on the presence of obstacles, roadworks or other vehicles in the rear blind spot. With that information, the in-vehicle application can instantiate both smooth **IoT-enabled speed adaptation and lane change** manoeuvres.

In such a way, IoT technology assists the rising of the automation level from 3 to 4.

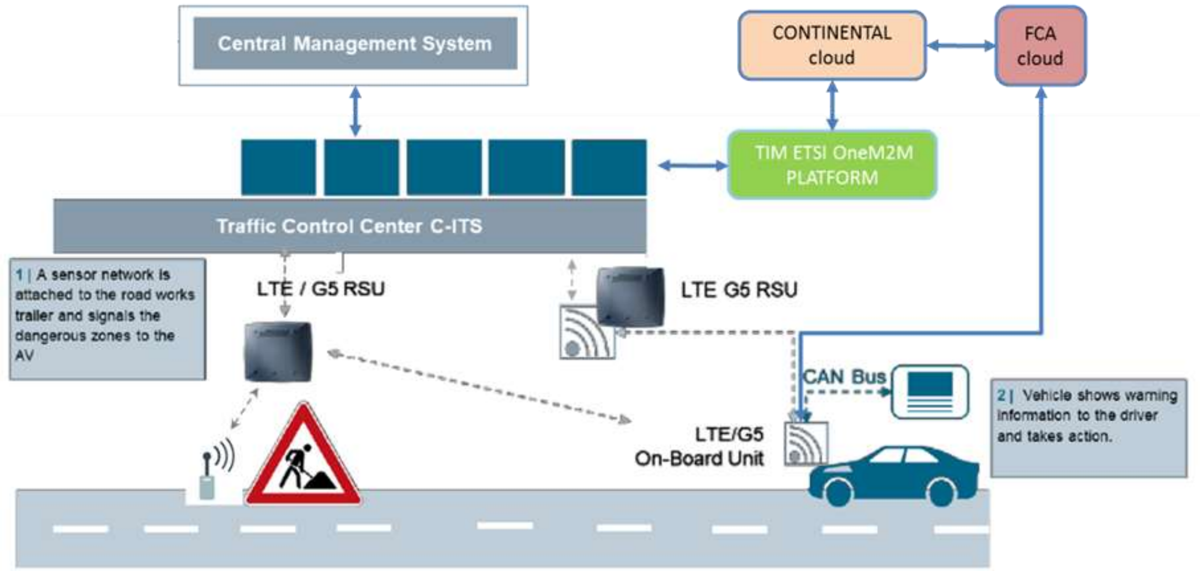

Figure 41 – Roadworks warning by TCC execution view

1.1.59 Urban driving use case functions (Livorno-Florence)

Urban driving use cases concern IoT-assisted speed adaptation in common urban scenario, considering traffic light, presence of bicycles, pedestrians and other vehicles. An overview of the execution is shown in Figure 42.

1. **Pedestrian detection with camera**
   - An AD vehicle is approaching an intersection regulated by a "smart" traffic light;
   - A smart camera detects a pedestrian or an obstacle on the lane. The information is processed locally and notified to the RSU via IoT protocols. Moreover, a connected traffic light sends to the RSU via SPAT/MAP messages information about the time-to-green/red;
   - The RSU receives the information transmitted by both devices (smart camera and traffic light), fuses the data and sends it by DENM messages to all the interested actors on the roads;
   - The OBU of the AD car receives the information and smoothly adapts the speed to the situation. The detection of VRUs and the traffic light status is also displayed on the MHI of the car;
   - The information is also sent to the OneM2M platform and can be retrieved by other vehicles in the same area via cloud applications;
   - At the same time the Port Monitoring Centre (MONICA) consumes the information from the oneM2M platform and displays a new advisory speed limit for the interested area to avoid possible problems.

2. **Connected bicycle**
   - An AD vehicle is moving in urban scenario with other road users, including a connected bicycle;
   - The connected bicycle is equipped with communication modules and dropout sensors: currently it sends CAM messages to other vehicles and to the infrastructure;
   - At a certain point, the bicyclist falls down while the AD car is approaching and a DENM is triggered;
   - The AD vehicle, informed by IoT of the dangerous situation, smoothly decreases its speed and stops before reaching the accident area;
   - The information is also sent to the OneM2M platform and can be retrieved by other vehicles in the same area via cloud applications;
   - At the same time the Port Monitoring Centre (MONICA) consumes the information from the oneM2M platform and displays a new advisory speed limit for the interested area to avoid possible problems.

3. **Potholes detection**
   - A wireless vibrations sensor installed on the vehicle notifies to the OBU, via 6LowPAN protocol, the occurrence of a vibrational shock above a certain level, due to a pothole presence on the road;
   - The OBU combines this information with other data coming from CAN bus (speed, odometer, etc.) and GPS and sends this data to the OneM2M IoT platform, by using CoAP and/or HTTP as application protocols;
   - An upcoming AD vehicle consumes the information and can arrange its speed accordingly

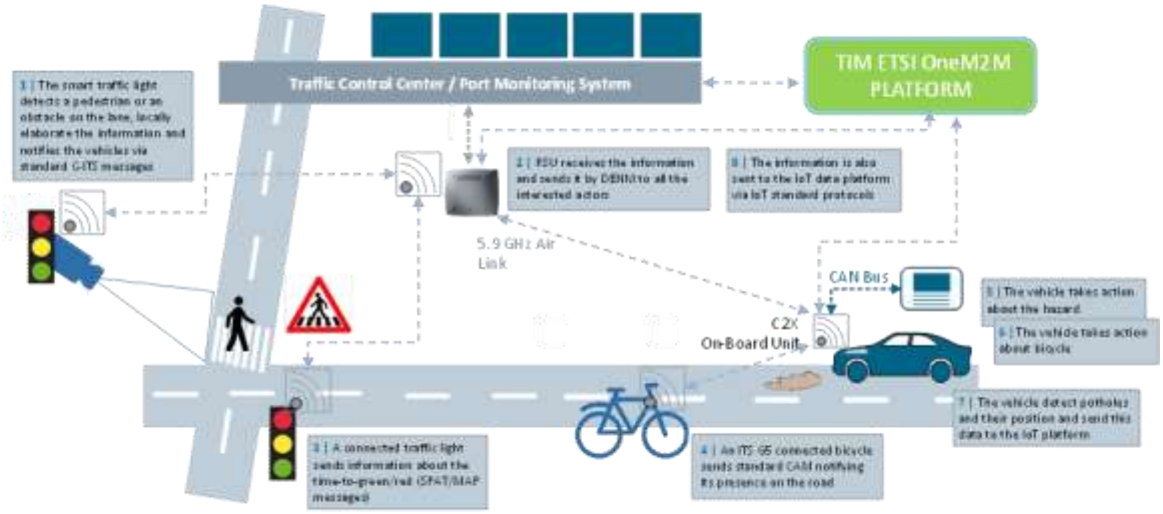

Figure 42 – Urban driving execution view

In such complex scenario, the IoT inputs to AD functions are many: IoT information about the traffic light phase and remaining time can be used from AD vehicles to adapt their speed in order to cross the intersection with green traffic light; and if not possible, to safety stop at the traffic light or queue behind other vehicles. Moreover, a smart camera on the test site can provide information on pedestrian traffic light violation. AD cars can use this information to stop at the traffic light even if the traffic light in its side is green. **IoT-enabled speed adaptation for AD car** is related also to the bicycle presence and if a fallen bicycle is detected, and to the road conditions. What is more, in this scenario, the IoT technology enhances the rising of the automation level from 3 to 4.

4.4 Spanish pilot site

The pilot site of Vigo is located in the north west of Spain. It is integrated in the urban section of SISCOGA corridor. It is extended along more than 100 km of urban and interurban roads in A55, A52, VG20 and AP9 highways.

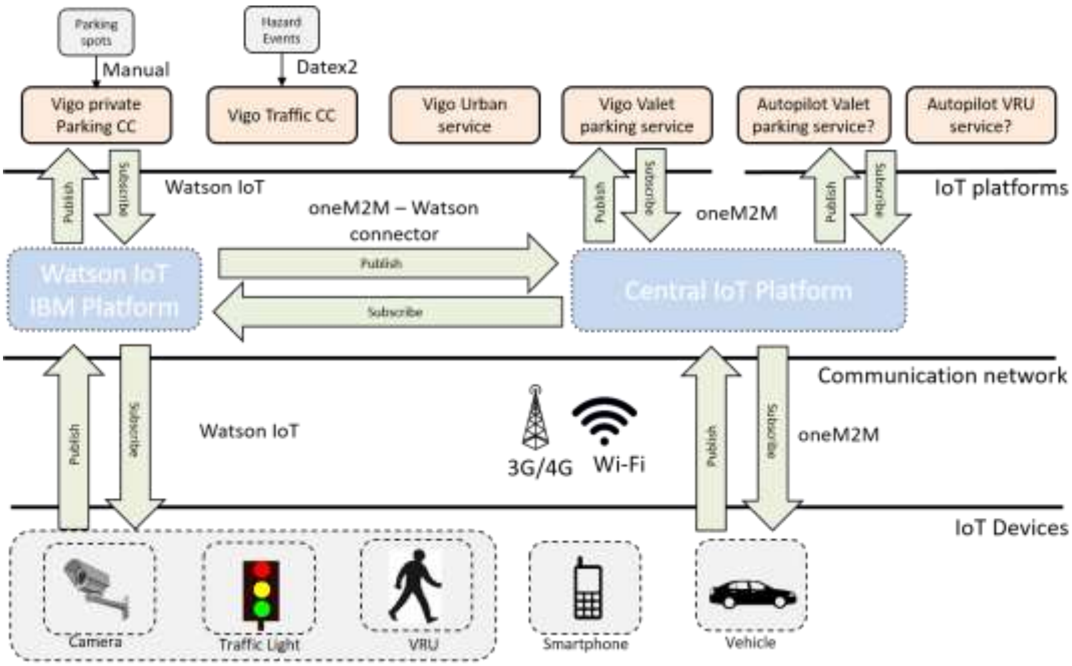

Figure 43 – Vigo pilot site architecture

The access to Vigo from AP9, A55 and VG20 are linked across the city roads through the main city streets. The SISCOGA facilities have enabled testing and development of multiple ITS solutions. Both urban and interurban sections are connected to management infrastructures. The Pilot Site architecture is detailed in Figure 43.

1.1.60 Automated valet parking use case functions (Vigo)

The aim of the Valet parking use case is to demonstrate how this functionality can benefit from different information sources, other than the on-board sensors, accessed via the principle of the Internet-of-Things like parking cameras. Through the use of IoT, the IoT platform can monitor and/or coordinate traffic on the parking lot and do efficient route planning based on real time available traffic information. Hence, the IoT platform will exchange information of the dynamic and static obstacles on the parking lot and/or the route to be followed by the vehicle using the information provided by the parking cameras.

Automated Valet Parking (AVP) has two main scenarios:

- Autonomously parking of the vehicle, after the driver has left the car at the drop-off point, which may be located near the entrance of a parking lot;
- Autonomous collection of the vehicle, when the driver wants to leave the site, he/she will request the vehicle to return itself to the collect point, using (for example) a smartphone app.

In Figure 44 the IoT devices and the functions that will be supported using the IoT platform are described. The following list is a detailed proposal of devices and functions to support the valet parking use case:

- Private parking control centre:
    - Informs when a parking spot is free or not;
    - Manages reservations;
    - Validates vehicle access;
    - Manages maps and vehicle routes;
- User’s mobile device:
    - Requests parking slots;
    - Manages pick-up and drop-off events;
- Smart cameras:
    - Publish detected events (e.g. pedestrians or other objects on the parking place);
- Connected AD car:
    - Validates that the access to the vehicle is provided to the authorized driver;
    - Informs when the vehicle is ready to move unmanned to the destination (parking place or collect point), e.g. when the driver has moved out of the proximity of the vehicle or has locked the doors;
    - Manages pick-up and drop-off events and Navigation to the destination, following a route either determined by the IoT platform, while avoiding obstacles detected by either the vehicle sensors or the IoT platform;
    - Informs when the car goes into a low power consumption mode;
    - Informs when an obstacle is detected;
    - Informs about vehicle sensors values and position.

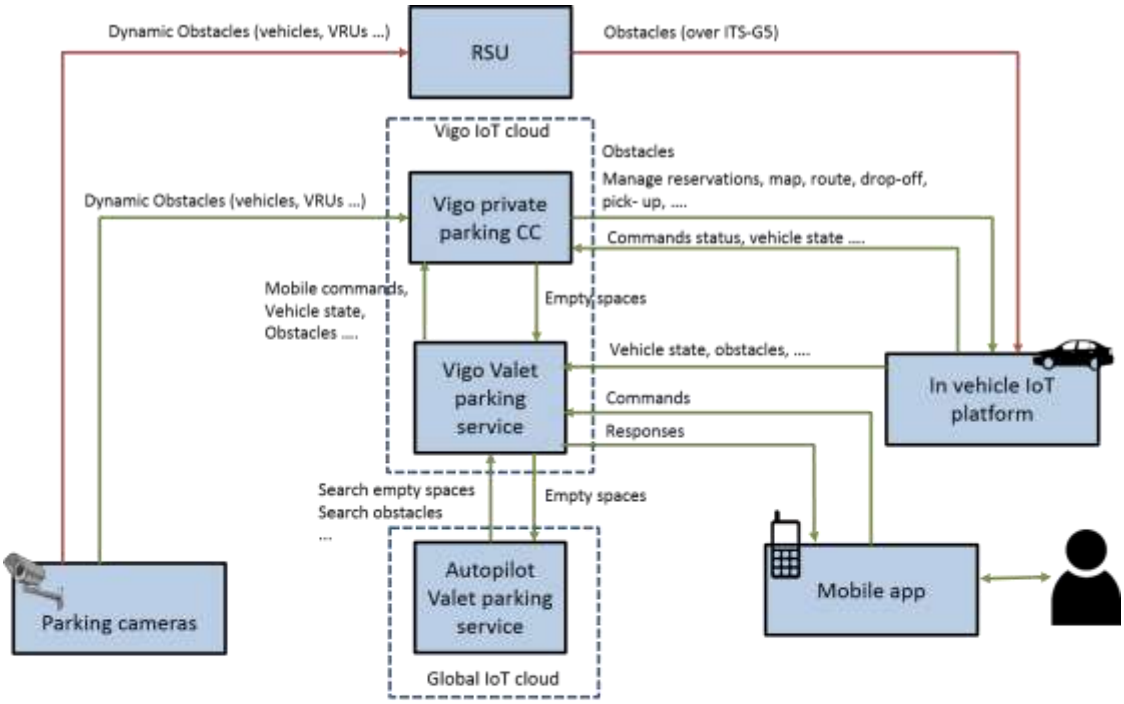

Figure 44 – Valet parking execution view

Interior parking areas are very controlled scenarios where the main challenges are the corners without visibility. IoT parking cameras can provide information on these blind spots, allowing the AD function to increase the automation level from 3 to 4.

1.1.61 Urban driving use case functions (Vigo)

Urban Driving assisted by IoT has the main objective to support CAD (Connected and Automated Driving) functions through the extension of the Electronic Horizon of an automated vehicle. The vehicle can process data from external sources that enrich those provided by its own sensors (Camera, LIDAR, Radar, etc.).
In Figure 44, the IoT devices and the functions that will be supported using the IoT platform are described. The following list is a detailed proposal of devices and functions to support the urban AUTOPILOT use case:

- Traffic control centre:
  - Informs when there is a hazard on the road (Accident, traffic jam, Road Work Warning);
- Traffic light:
  - Informs about the traffic light status and time to change;
- Smart cameras:
  - Publish detected events (e.g. pedestrians or other objects on the intersection).
- Connected AD car:
  - Informs when an obstacle is detected;
  - Informs about vehicle sensors values and position.

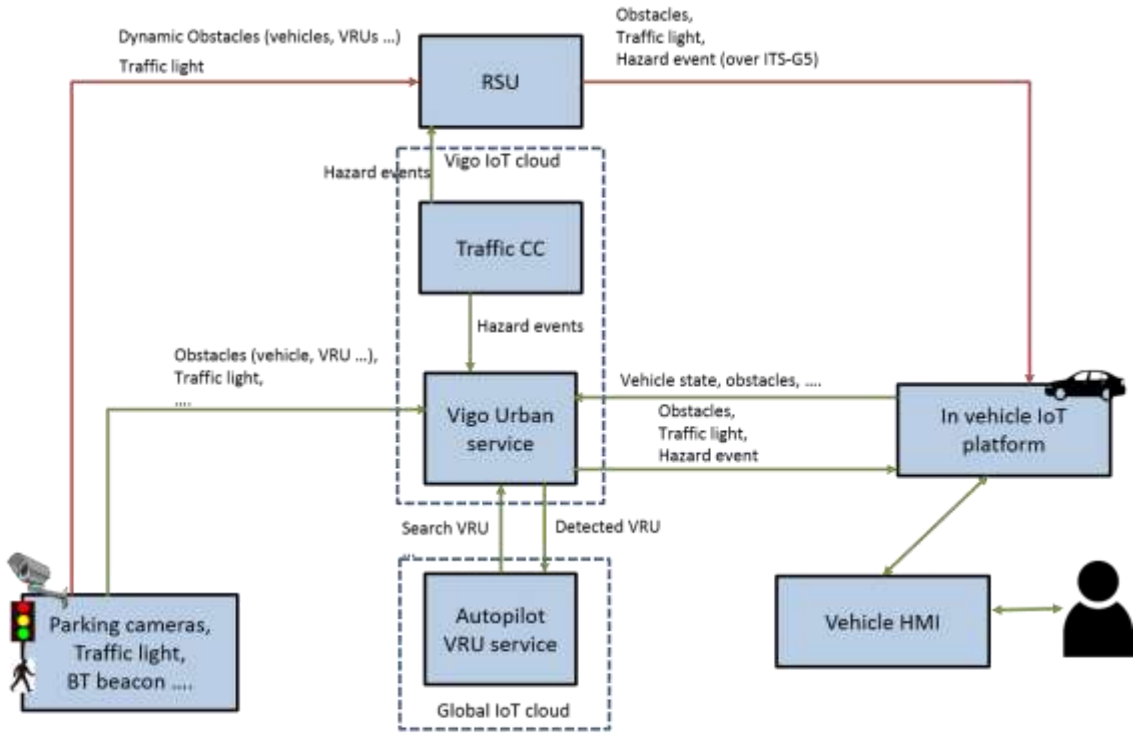

Figure 45 – Urban AUTOPILOT execution view

Considering all the information provided by IoT devices, the CAD systems will adapt their behaviour accordingly.

The complexity of urban scenarios makes it essential to have as much redundancy information as possible. IoT platform provides data about the traffic lights and road events through 3G/4G.

Furthermore, the frequency with which the IoT platform sends the data is higher than other advanced V2X communication. For the case of VRUs, the information received by IoT complements the data from the AD sensors, so it provides more reliable and accurate results.

There are other objects that could not be detected if there was no IoT Services (IoT Camera or sensor information of other cars). As a result, IoT technology allows increasing the automation level from 3 to 4.

4.5 Finnish pilot site

The pilot site in Finland is located in Tampere, in the town district of Hervanta, at the premises of VTT and on the public roads in the neighbourhood.

The major road (Hervannan Valtaväylä) connecting Hervanta to the city centre is a road with two lanes in each direction, with maximum speed limit of 50 km/h.

There is a separate cycle track at the east side of the road. The vehicles that will be used in the test site are research vehicles from VTT, e.g. a Citroen C4 that has been converted by VTT for automated driving and acts as an innovation environment where industry can test sensors and applications.

VTT has also a mobile road side unit on which roadside infrastructure can be installed. For the AUTOPILOT use cases, a traffic camera is installed on the mobile road side unit.

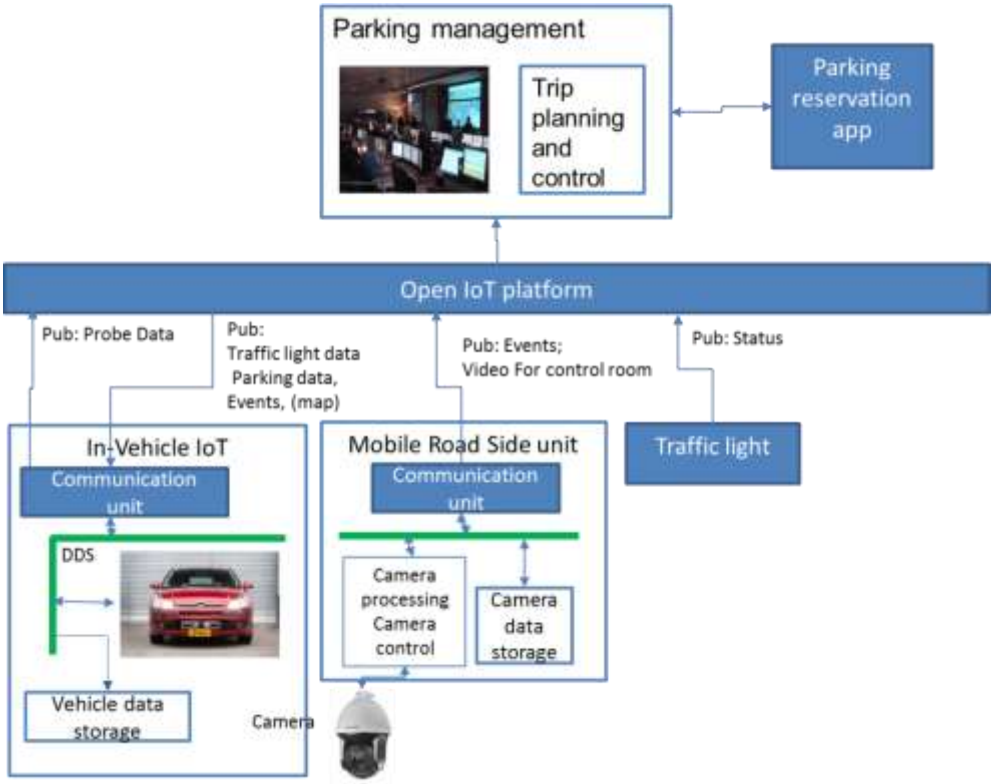

Figure 46 - Architecture of the Finnish pilot

1.1.62 Automated valet parking use case functions (Tampere)

The main scope is to show how automated valet parking can be realized, and how it can benefit from the use of IoT technology. The intention is to provide the vehicle with additional information that increase the automatic valet parking capabilities based on IoT technology, i.e. information that cannot be sensed by the vehicles environmental perception sensors. A traffic camera will provide information regarding obstacles that can affect the planned route. The inclusion and use of IoT technology will move the automation level from 3 to 4, limited to the operational design domain (ODD) automatic valet parking allowing driverless parking of the vehicle.

The Automated Valet Parking in Tampere demonstrates the drop-off phase. A link to a reservation system is provided, developed within the framework of the Transforming Transport. Prior to arriving at the drop-off point, a reservation parking place near the drop-off point is reserved. At the drop-off point, the parking management system takes over control of the vehicle, and the vehicle starts moving in unmanned mode to the reserved parking space. Traffic cameras detect the availability of the parking space and presence of objects on the path.

1.1.63 Urban driving use case functions (Tampere)

The main scope is to show how intersection support can be realized in urban driving, and how it can benefit from the use of IoT technology. The intention is to provide the vehicle with additional information that increase the urban driving capabilities based on IoT technology. For instance, the IoT-enabled traffic lights will give the vehicles access to real-time information on the traffic light status, and a traffic camera will provide information about potentially conflicts with vulnerable road users (VRUs). In case the traffic camera detects a VRU that will pass during the same green phase as the vehicle, which turns over the path of the VRU, the vehicle will be warned. The inclusion of IoT technology in the roadside infrastructure and the V2I communication will contribute to a step forward towards autonomous urban driving.

5. IoT platforms and IoT devices integration

IoT devices for autonomous driving applications are deployed through IoT platforms that offer integrated service where the IoT devices interact and exchange information. Chapter 5 describes the integration of IoT devices into IoT end-to-end platforms. These provide the hardware, software, connectivity, security and device management tools to handle the different IoT device used in the different use cases across the AUTOPILOT pilot sites. Different sections provide information on how the integration is implemented presenting the managed integrations, device management, cloud connection, cellular modem, etc., to manage and monitor the IoT devices in different use cases.

5.1 IoT platform and IoT devices integration - Versailles

The oneM2M standard defines two mechanisms to integrate oneM2M and non-oneM2M IoT devices into the IoT platform:

- Integration of oneM2M devices: In this case, IoT devices are called Application Dedicated Nodes (ADN) and can interact with the oneM2M platform directly via the MCA interface. The IoT devices send requests and receive notification using the oneM2M RESTful API;
- Integration of non-oneM2M devices: the oneM2M standard is highly extensible and allows the integration non-oneM2M devices and applications, regardless of their vendor or provider. A dedicated software component called Interworking Proxy Entity (IPE) shall be developed and deployed for this purpose. The IPE provides interworking between oneM2M platform and specific IoT device technologies or protocols.

Figure 47 shows main components and interactions of the Versailles pilot site. The connectivity within the vehicle is handled by a Vehicle Connectivity Module developed by NEC.

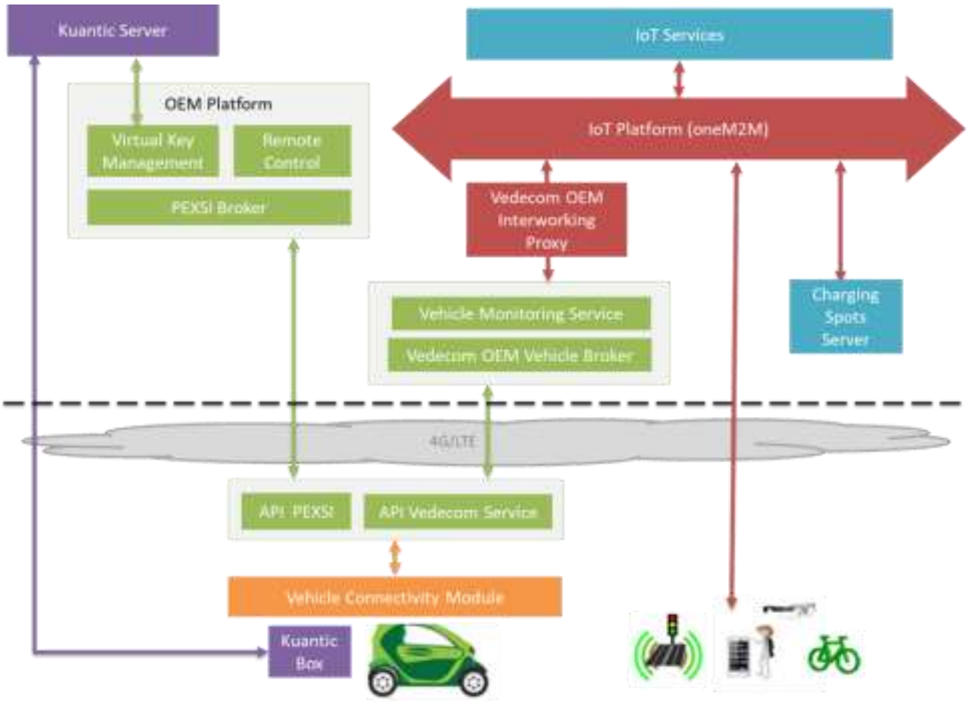

Figure 47 – IoT platform and IoT Versailles integration

Vehicle remote control data is not exposed to the IoT platform. It is pushed to OEM Platform (via the PEXSI Broker) using a separate interface (API PEXSI) available on the vehicle. In addition, a separate communication channel for virtual key management is established between Kuantic Server, deployed on the cloud, and Kuantic Box, deployed on the vehicle.

Vehicle monitoring data is exposed to the Vedecom OEM vehicle Broker using the API Vedecom Service available on the vehicle. Data is pushed to the IoT Platform via an Interworking Proxy Entity to make it available for high-level IoT services using a generic data model and MCA interface.
Other IoT devices including traffic lights, bicycles, charging spots, passengers' devices are considered as oneM2M-enabled devices and will interact with the IoT Platform using MCA interface.

5.2 IoT platform and IoT devices integration - Brainport

There are several use cases that will be implemented and rolled out on the Brainport pilot site. One must understand that the use case implementations are being developed by various project partners using different IoT platforms and technologies. In case of Brainport, there are plans to use:

- OneM2M Interoperability Integration Platform from Sensinov (exploitation by TNO)
- FIWARE IoT platform from NEC
- Watson IoT platform from IBM
- Huawei IoT platform

Since the platforms generally perform similar tasks and provide comparable interfaces (e.g. device management, like discovery, message brokers, etc.), there is a challenging task to make all components work together. Moreover, the pilot site devices should be able to connect to one of the platforms and platforms should be able to discover devices and communicate with them.

The goal is to make platforms and devices interoperable. Figure 48 – Target platform integration shows a proposed integration between platforms and devices. On this figure, there are:

- AUTOPILOT applications that implement use cases;
- An instance of oneM2M platform from Sensinov. By default, devices should connect to this platform and it should hide the complexity of the communications between platforms and applications;
- A set of IoT platforms that should either able to communicate with the oneM2M platform or implement support oneM2M communication protocols by itself;
- Pilot Site IoT devices connected to the oneM2M platform.

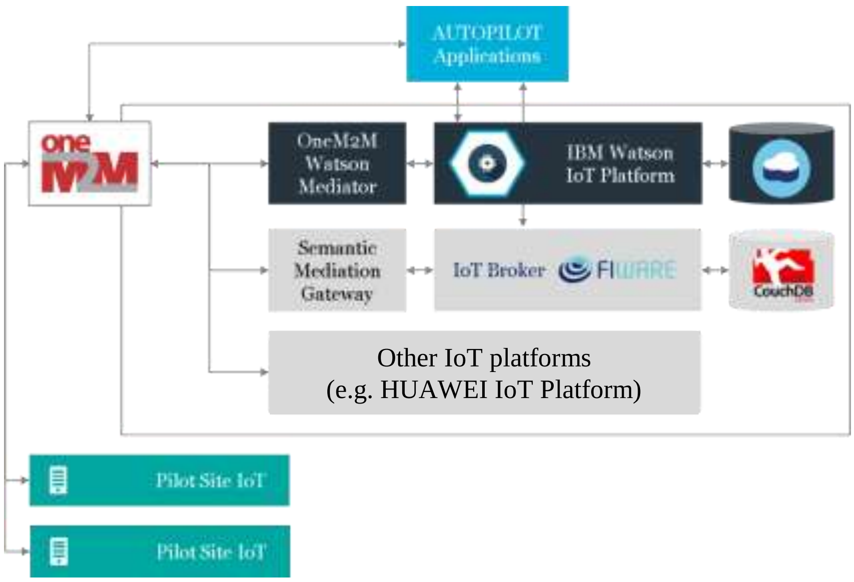

Figure 48 – Target platform integration

In this scenario, there are two platforms involved in the communications from bottom up to the top: device, oneM2M interoperability platform, then the target IoT platform and finally an application that deals with the device. The same stack is unwinded backward when the application sends a message to the device. So, the process looks a little more complicated than it could be.

To reduce the burden of the interoperability between platforms, there may be another case where a device connects directly to the target IoT platform, e.g. IBM Watson IoT platform. This may be useful if one knows that messages from the device will be consumed only by one IoT platform. In this case, there is no need to build a hierarchy of the platforms and pass the emitted by the device messages though the full stack. The drawback of this approach is that the interoperability platform doesn't know all the connected devices. To address this problem, an "announcement" process may be introduced. When a device is connecting to an IoT platform, this platform makes an announcement to the interoperability platform saying that this certain device is connected to the certain IoT platform and if somebody wants to consume data from the device via the interoperability platform, it must look up for the device and message at this IoT platform.

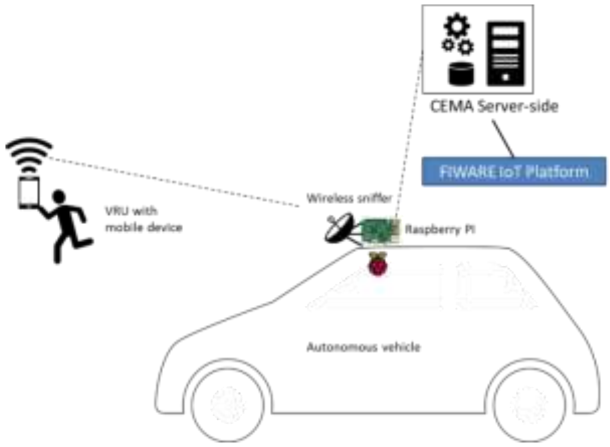

Figure 49 – Wireless sniffer integration with FIWARE IoT Platform.

The above figure shows an overview for the **integration of the wireless sniffer device** to the FIWARE IoT Platform.

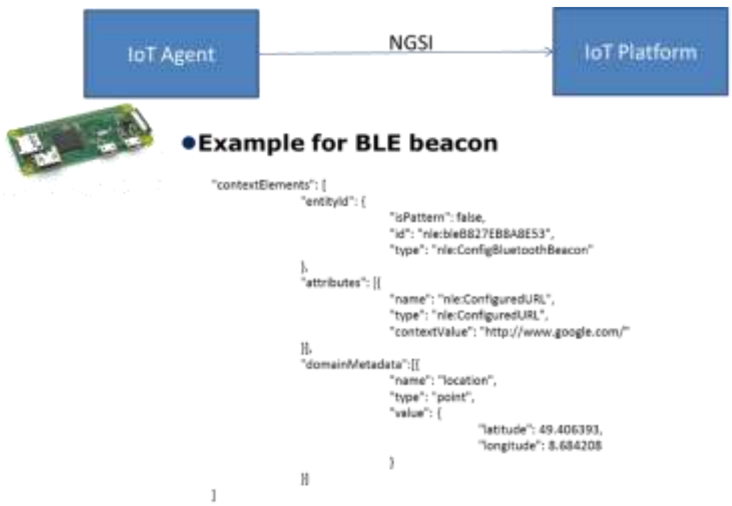

Figure 50 – Integration of BLE beacon to the FIWARE-based IoT Platform.

The wireless sniffer, after receiving the Wi-Fi probes from a VRU, sends the data through 3G or 4G communication to the CEMA server-side component, which is a FIWARE-based integrated component that operates using the IoT Broker of FIWARE IoT Platform. The information model of the CEMA is NGSI-based and it is explained in detail in terms of its structure and semantics.

The above figure illustrates the mechanism to **integrate a BLE beacon** to the FIWARE-based IoT platform. A BLE beacon (Raspberry Pi Zero W) has software called "IoT Agent", which implements the Next Generation Service Interfaces (NGSI) for registering itself to the IoT platform as well as sending updates to or queries from the IoT platform. The NGSI interface is a REST-based interface and NGSI data model can be seen in the message body. In this JSON object, the information of the BLE beacon such as the ID of the BLE beacon, the broadcasted URL ("nle:ConfiguredURL") as well as the location ("latitude", "longitude" values) can be pushed to the IoT platform.

See section 0 for more complete overview of platforms and IoT interconnections.

5.3 IoT platform and IoT devices integration - Livorno-Florence

At the Italian pilot site, the IoT devices are integrated in the IoT oneM2M platform according to the oneM2M standard, as shown in Figure 51.

- IoT oneM2M platform: is a federated model where several heterogeneous IoT platforms are interconnected. A central IoT platform includes various modules: big data management and storage, real time and batch analytics, security and privacy, semantics, etc. Interoperability between the central IoT platform and the Pilot site IoT platforms is addressed in this platform. More detailed information about this platform is available in the deliverable D2.3 (Report on the implementation of the loT platform);
- In-vehicle IoT platform: is an in-vehicle component that provides a communication with the Cloud IoT platform and the interfaces to other in-vehicle components. More detailed information is available in the deliverable D2.1 (Vehicle IoT integration report).

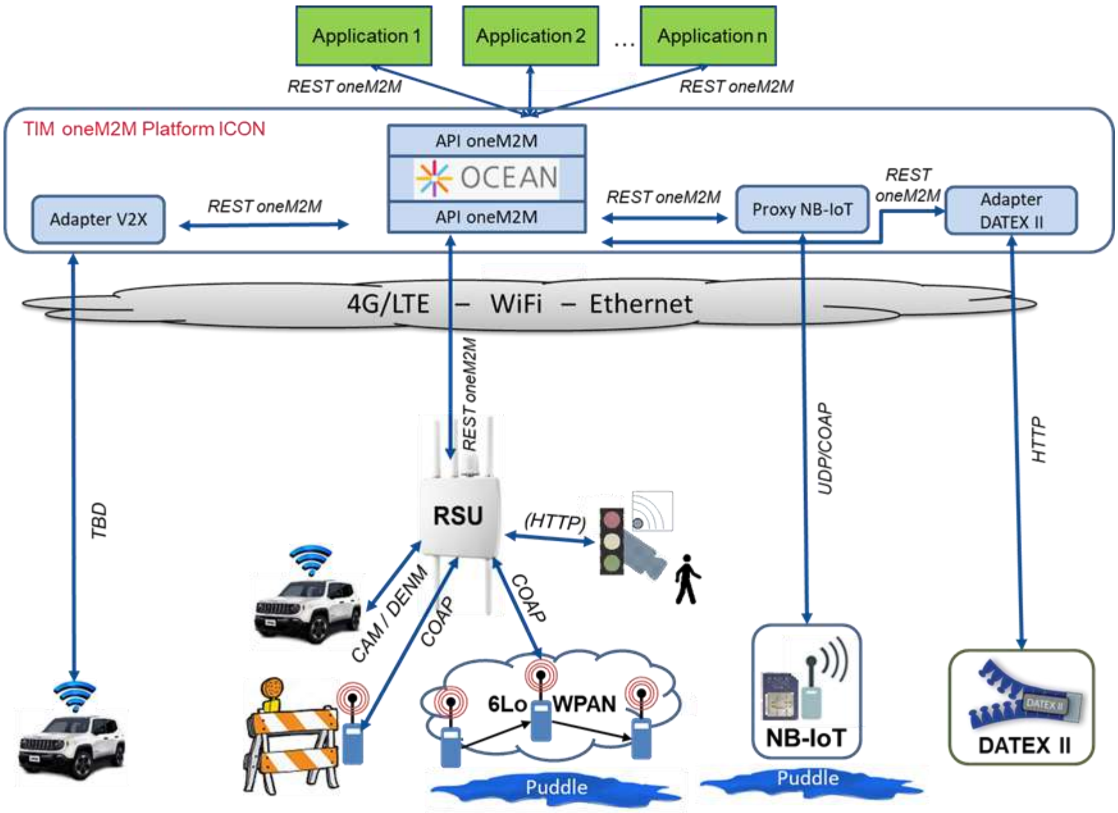

Figure 51 – Integration of IoT devices into the oneM2M IoT platform at the Italian PS

5.4 IoT platform and IoT devices integration - Vigo

The Vigo pilot site is composed by three main IoT platforms:

- **IoT platform**: is a federated model where several heterogeneous IoT platforms are interconnected. A central IoT platform includes various modules: big data management and storage, real time and batch analytics, security and privacy, semantics, etc. Interoperability between the central IoT platform and the Pilot site IoT platforms is addressed in this platform. More detailed information about this platform is available in the deliverable 2.3.

- **In-vehicle IoT platform:** is an in-vehicle component that provides a communication with the Cloud IoT platform and the interfaces to other in-vehicle components. A more detailed information is available in the deliverable 2.1 Vehicle IoT integration
- **Device/s IoT platform:** the devices can be new devices or existing devices adapted to become IoT devices able to be integrated into the IoT ecosystem. More detailed information is available in detail in the following paragraphs.

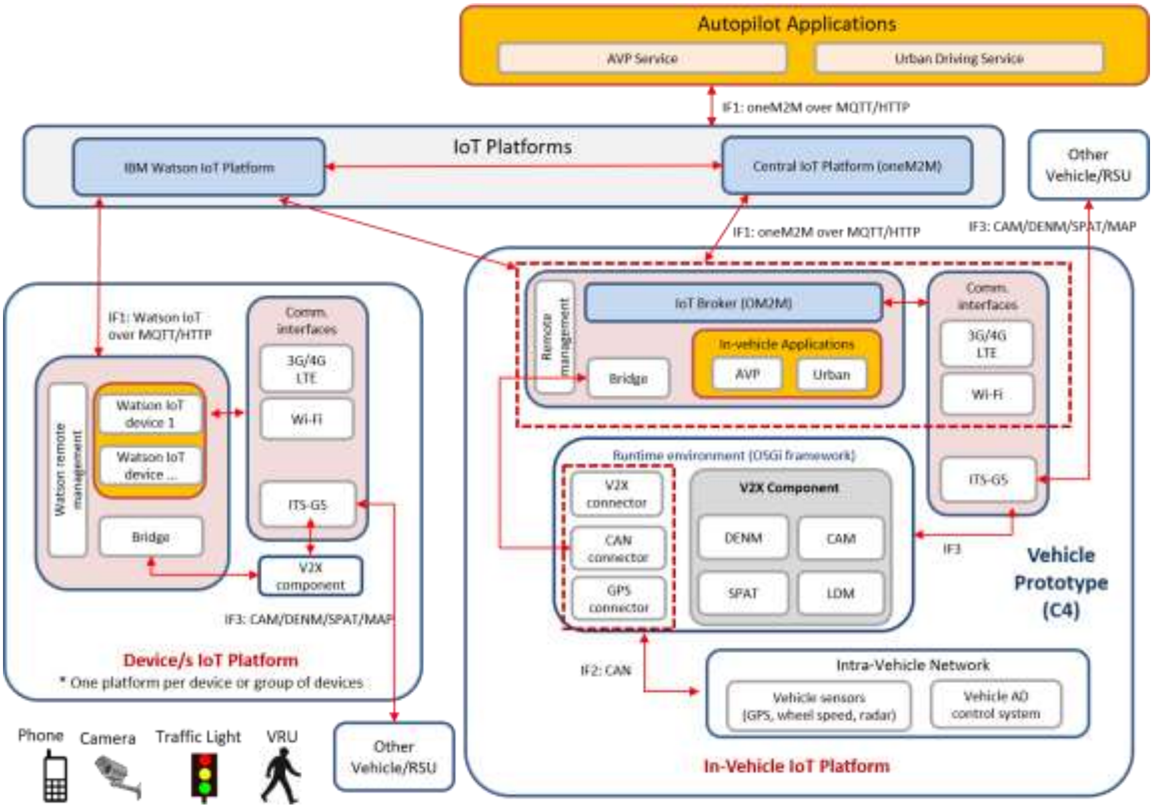

Figure 52 – Vigo IoT platform and IoT devices integration

The in-vehicle platform and device platform software components are described in Figure 52. The more important software components are:

- **Communication interfaces**: is the component responsible for providing connectivity to the device. The supported interfaces are cellular (3G/4G LTE), Wi-Fi and ITS-G5 wireless interface;
- **IoT Module**: This module translates the information that comes from the different devices into oneM2M messages and translates any oneM2M message into understandable information for the vehicle;
  - **IoT Broker**: OM2M based ASN-CSE (oneM2M) that acts as an IoT Gateway. It provides the HTTP and MQTT connectivity to the Cloud IoT platform;
  - **Bridge**: is responsible for translating all the information from the vehicle into oneM2M and for publishing and providing any needed methods to obtain this data;
  - **IoT Applications**: are responsible for the interaction with the physical devices in order to provide the full functionality expected in the use cases;
- **Runtime environment:** OSGi framework that contains the stack that enables the V2X communication;

- **V2X Component**: Contains several modules that are able to process data coming from V2X communication through ITS-G5. Provides the encoding/decoding for the SPaT/MAP, CAM and DENM messages. Includes the connectors that give access to the IoT module.

5.5 IoT platform and IoT devices integration - Tampere

Figure 53 shows the architecture of the Finnish pilot. The pilot has the following platforms:

- An **open IoT platform** for connecting the different devices based on oneM2M. The main purpose of the IoT platform is to act as a broker. The platform is described in more detail in deliverable 2.1;
- **In-vehicle IoT platform** provides communication with the IoT platform and with the different devices and applications in the vehicle. Data is exchanged between the different applications using DDS (Data Distribution Service). More information is provided in deliverable D2.3;
- **Mobile road-side unit** has a similar architecture as the vehicle. The mobile road side unit processes the information from the traffic camera and makes this information available through the IoT platform to the vehicle and the parking management system. The system also has storage process for assuring that all data needed for evaluation are made available;
- In addition, there is a connection to the **traffic light server**. Information on the traffic signal phases is available in real time both through ITS-G5 as standardised SPAT/MAP messages and over cellular as MQTT messages.

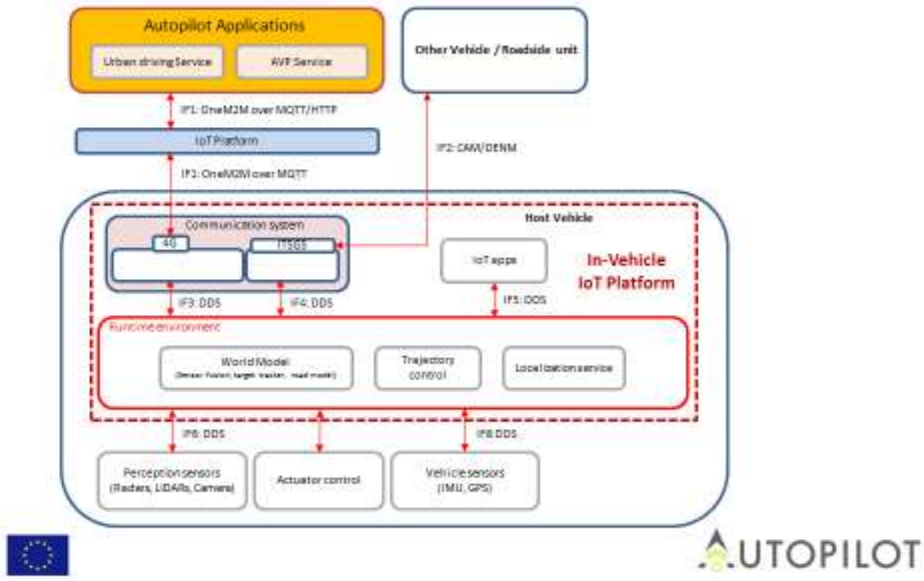

Figure 53 – Architecture of the Finnish pilot

6. Communication, security and platform interfaces

Chapter 6 describes the communication and connectivity implemented in the different use cases using the IoT devices developed for providing different functions. The description address as well the IoT platforms interfaces used and the security mechanisms implemented for the different specific cases and pilot sites.

6.1 Dutch pilot site

The communication infrastructure of the Brainport pilot site is fairly complex as it involves multiple services, platforms, devices and corresponding logical and physical communication interfaces. The diagram of the technical architecture emphasizing the physical communication interfaces is depicted in the figure below.

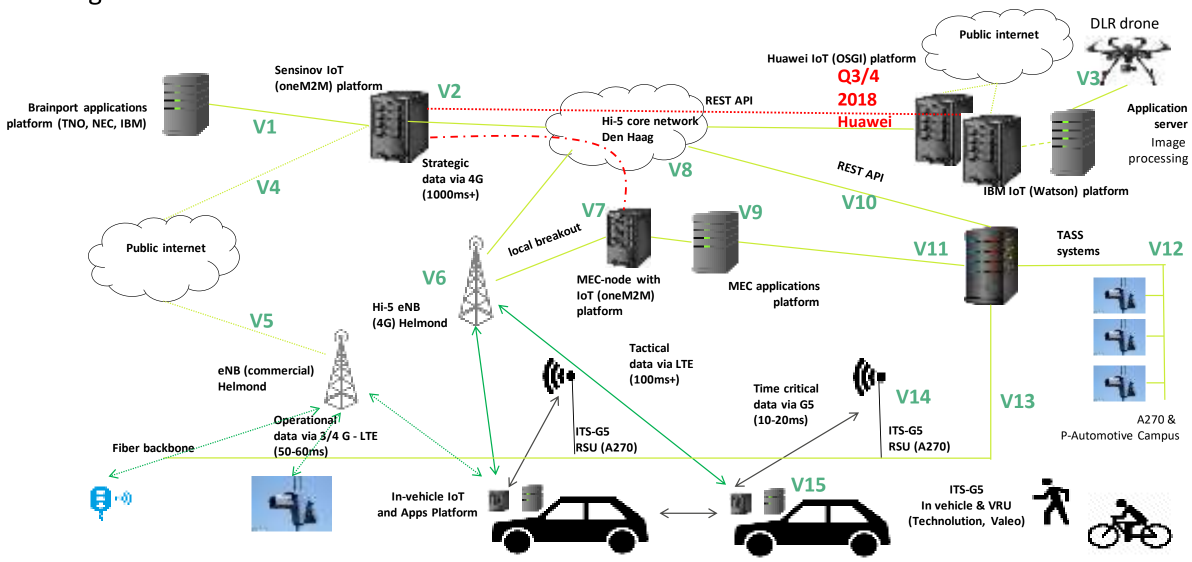

Figure 54: Technical architecture Brainport, emphasizing the physical connections

Below a table provides an overview of all relevant interfaces with brief specification, description and current status:

Table 12 - Overview relevant interfaces

Code	A-side	B-side	Type	Physical connection	Brief description	Current status
V1	Sensinov IoT	Applications NEC, Huawei, IBM	MCA	http, mqtt, ngsi, watson	IoT interaction with functions from other platforms (applications)	Active
V2	Sensinov IoT	Technical infra	IP	Connections datacentre	Physical infrastructure	Active
V3	MAV device	IBM platform	Watson	http, mqtt	MAV device as IoT	Active
V4	Sensinov IoT	Public Internet	IP	Connection to internet	Physical / cloud	Active
V5	Internet to eNB commercial	Public telco	IP	Telco	Connectivity	Active

V6	Sensinov IoT	Hi5 eNB dedicated as preparation for MEC applications	IP	Dedicated IP using private infra	Preparation for MEC in further technical tests	Active
V7	Sensinov IoT	TASS technical platform	IP	Secured IP connection (private>public> private)	MEC computing local breakout	Not tested
V8 <- V10 -> V9/V11	Dedicated TNO cloud	TASS technical platform	IP	Dedicated IP using private infr	Brainport technical ICT	Active
V12 – V13 – V14	Dedicated road side infrastructure	Vehicles / Devices	(Wi-Fi) IP	IP / Wi-Fi-p	Cooperative V2X roadside infrastructure	Active
V15	Road side infra	(In vehicle) IoT devices - TLC - Vehicle - Cameras	MCA	IP / Wi-Fi-p	IoT devices connected	Active Active Active Active

The various IoT and application platforms are interfaced as depicted below.

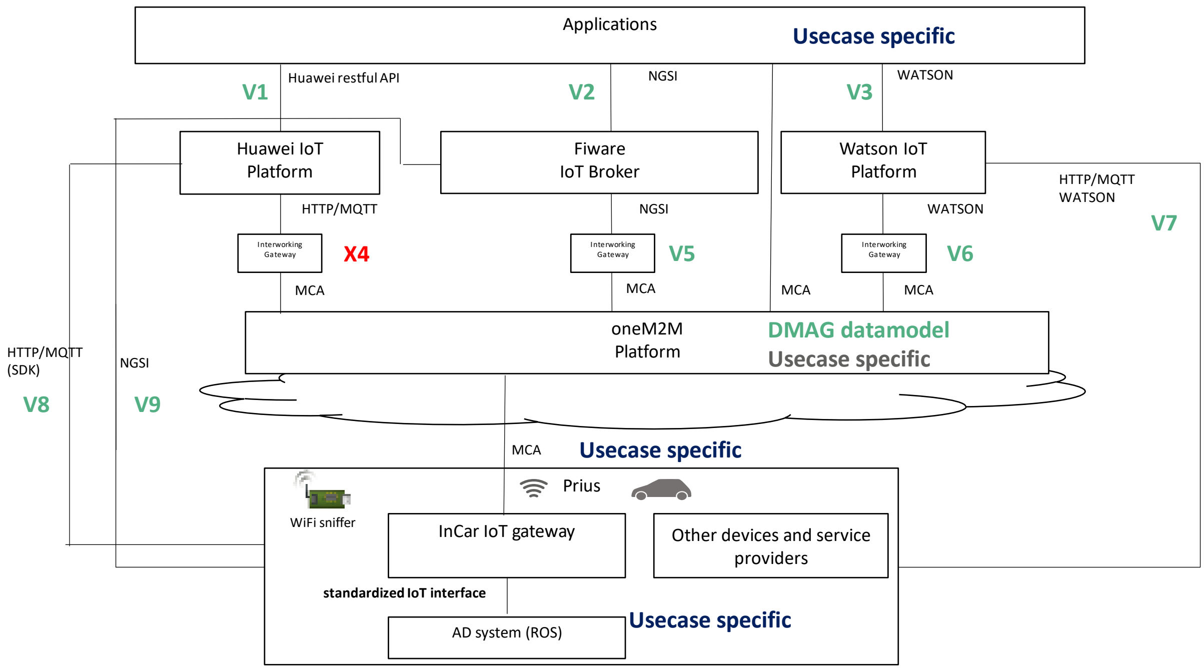

Figure 55 - Platform interfaces in Dutch Pilot site Brainport

Below a table provides an overview of all relevant interfaces with brief specification, description and current status:

Table 13 - Overview relevant interfaces

Code	A-side	B-side	Type	Physical connection	Brief description	Current status
V1, V2, V3	IoT platforms	Applications provided by IoT platforms	MCA	RESTful, NGSI, Watson	Suppliers of applications provided through IoT platforms	Active
X4	Huawei IoT	Sensinov IoT	MCA	Http, MQTT	Interoperability between	Not active

					platforms	
V5	NEC Fiware	Sensinov IoT	MCA	NGSI	Interoperability between platforms	Active
V6	IBM Watson	Sensinov IoT	MCA	Watson	Interoperability between platforms	Active
V7, V8, V9	IoT platforms	Device	Proprietary	Http, MQTT, NGSI, Watson	Direct connections to IoT devices	Active
V10	Sensinov IoT platform	Device	MCA	Http, MQTT,	Connection IoT devices with Brainport OneM2M Sensinov	Active

1.1.64 Communication interfaces

The information regarding the communication interface of the FIWARE IoT Platform are placed in the D1.3 (Initial IoT Self-organising Platform for Self-driving Vehicles) document of IoT architecture specification (task T1.2) as well as D1.7 (Initial specification of Communication System for IoT-enhanced AD) document of communications specification (task T1.4). Same information is placed below; however, more information can be easily found in the detailed specifications documents of NGSI9 and NGSI10 interfaces, which are published by FIWARE. FIWARE focuses on a common data model and powerful interfaces for searching and finding information in IoT. FIWARE is using the OMA Next Generation Service Interface (NGSI) data model as the common information model of IoT-based systems and the protocol for communication. NGSI-9 and NGSI-10 are HTTP-based protocols that support JSON and XML formats for data. Let us shortly describe these two interfaces.

NGSI9: it is used to manage the availability of context entity. A system component can register the availability of context information, and later on the other system component can issue either discover or subscribe messages to find out the registered new context information. Detailed specifications can be found in [22].

NGSI10: it is used to enable the context data transfer between data producers and data consumers. NGSI10 has query, update, subscribe and notify context operations for providing context values. A context broker is necessary for establishing data flow between different resources as well as consumers or providers. Detailed specifications can be found in [23].

The MAV and its ground station computer act as one single IoT device in the AVP case at Brainport pilot site in the Netherlands. The communication between the MAV and the ground station is based on a local IEEE 802.11n Wi-Fi connection that guarantees high-data bandwidth and continuous local availability. Small Open Mesh OM2P routers are used on both sides. The ground station PC connects to the Watson IoT platform via 4G/5G.

6.2 Italian pilot site

IoT solutions require that environment or device status be made available to a cloud-based application for consumption by a variety of stakeholders.

Goal of the Italian Pilot Site use cases of Highway and Urban Driving is to demonstrate how additional IoT sensors placed in the AUTOPILOT prototype can enhance the functions of the car itself. In such a way, the vehicle can be used for example as an IoT sensor for detecting the surface condition for both highway and urban scenarios. In order to satisfy the different Use Cases of the

Italian Pilot Site (Highway and Urban Driving), the vehicle needs an on-board IoT platform (OBU) to handle the various sources of data (IoT sensors like Inertial sensors, Lidar, Camera, IMUs, etc.) with the various services (LDM, Pothole Detector, etc.).

As showed in different schemes in this chapter, the IoT solution is characterized by devices (i.e. OEM in-vehicle components, inertial sensors, smartphones) that use a gateway (On Board Unit) to integrate the IoT information and to communicate to a Cloud server based on OneM2M platform.

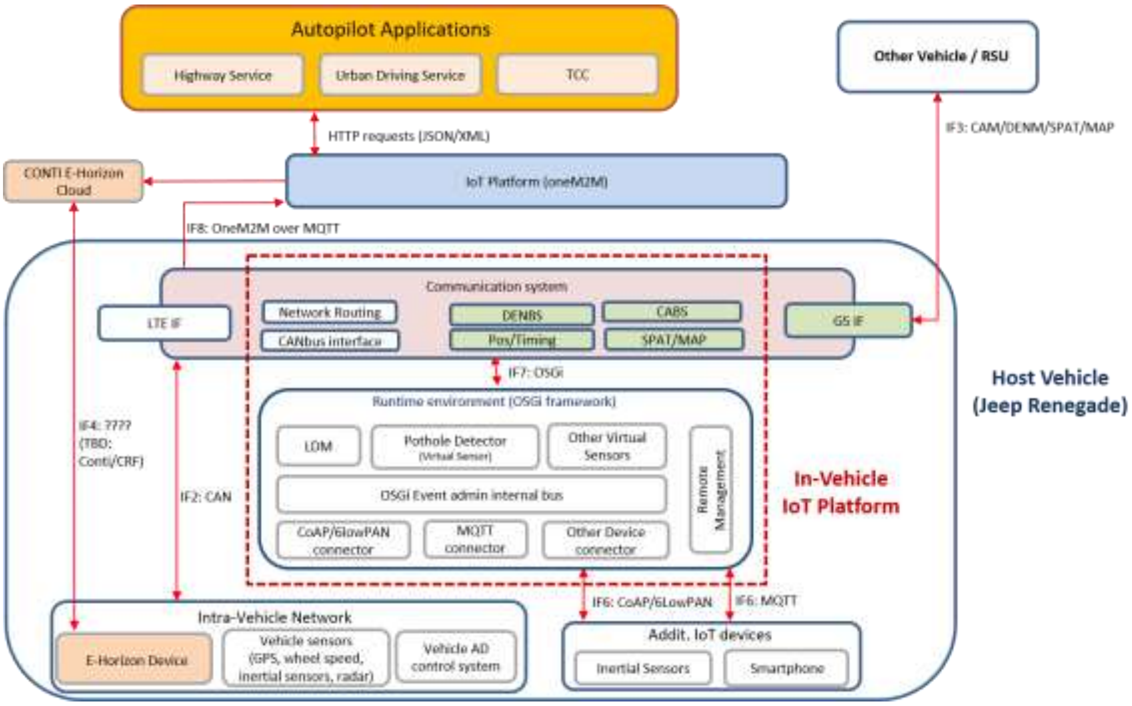

Figure 56 – In vehicle software architecture for IoT platform integration (Italian pilot site)

The in-vehicle software architecture for the IoT platform integration of the Italian Pilot Site is showed in Figure 56. In this scheme, it is possible to represent the IoT in vehicle Platform and also the interconnections between this container with other on-board sensors in the host vehicle (e.g. Jeep Renegade) and the cloud system and/or other vehicle/Road Side Unit.

Figure 57 – ISMB IoT in vehicle platform

ISMB IoT In-Vehicle platform is composed by a quad-core ARM processor supplied by ROJ. The platform runs an optimized version of Linux and provides several interfaces like: IEEE 802.11p (ETSI ITS-G5, Wi-Fi, BT, Ethernet, CAN, 6LoWPAN, LTE. The IEEE 802.11p (ETSI ITS-G5) connectivity is provided by a Qualcomm module. All of the ETSI ITS-G5 stack is developed by ISMB and can be adapted to the needs of the project. The board implements CAM, DENM and SPAT/MAP standards with the possibility to send messages both over IEEE 802.11p and LTE channels. The board will manage the lane level computation of the surrounding vehicles position. Finally, it mounts a GNSS (Galileo + GPS) receiver that is used for positioning and synchronisation. The unit can synchronize other hosts using the NTP standard or other protocols.

The IoT in-vehicle platform of the Italian Pilot Site is modular software including Application Container and Communication System, which are deployed on the On Board Unit (OBU). The "Runtime Environment" part of the OBU is composed by several software modules.

The functionality of Remote Management is implemented by a software (**OSGi remote management tool**), which allows to configure the platform by adding/removing bundles, introducing the idea of remote monitoring and control of external application based on OSGi platform. Through the Event Admin Internal Bus, the connectors have the same communication interface to the bundles, which they interfaced in the Application Container.

The Application Container also encases the functionality of Data Management, with the modules of Local Dynamic Maps (LDM) and the Pothole Detector. **LDM** is a database that achieves integrated management of map information and vehicle information (functional requirement of Context Awareness): it contains information on real-world and conceptual objects that have an influence on the traffic flow. The bundle of **Pothole detector** represents the implementation of the pothole detection algorithm. It is based on data fusion techniques in order to implement the concept of "virtual sensors". This module collects data from multiple sensors on the vehicle (IoT in-vehicle components or OEM in-vehicle components), processes the various data and sends the results of this elaboration to the cloud OneM2M platform or RSU or other vehicles (via communication system).

Regarding the IoT device adaptation, it is planned to support different IoT communication protocols with the devices: the IoT connectors showed in the Figure 56 are used to integrate with 6LoWPAN data coming from additional IoT devices (i.e. Inertial sensors), which are used by edge applications on the OBU (**CoAP/6LoWPAN connector**). They are also used to integrate with MQTT protocol data coming from additional IoT devices (i.e. smartphone), which are used by edge applications on the OBU (**MQTT connector**).

The "Communication System" part of the OBU manages different high-level capabilities. The module **CANBus Interface** reads data coming from the CAN Bus and decodes important data coming from the in-vehicle sensors that are sent directly to the OneM2M platform or used by edge applications on the OBU.

The module **Pos-Timing** reads the positioning data and timing information through the GPS hardware module in order to set the position on CAM and DENM messages.

CABS and **DENBS** modules take data from the CANBus, position and time from Pos-Timing and create a CAM/DEN message as described in the proper ETSI standard [9]. They also receive CAM/DEN messages coming from other vehicles and save them on the LDM.

The **SPAT/MAP** messages in the Communication System are generated from a traffic light and SPAT/MAP module decodes them, saving the relevant information in the LDM for further use. SPAT/MAP offers a potential channel for detailed information exchange between traffic systems and road users.

The capability of message routing is assigned to **Network routing**, which manages the connectivity of all the in-vehicle modules that need network connectivity. Moreover, it manages the channels where CAM and DENM messages are sent. In the ISMB OBU, they can be transmitted on the ETSI G5 radio channel and/or on the cellular way for debugging or other purposes.

As far as the interoperability part is concerned, it should be considered that the in-vehicle IoT platform should work with heterogeneous devices, technologies, applications, without additional effort from the application or service developer.

OEM-specific components relate to components such as actuators for power steering and brakes, inputs to gearbox or vehicle sensors needed for the "normal" vehicle functions (MAP, MAF, ABS, etc.). Software modules implementing drivers to virtualize such OEM-specific components into Vehicle IoT Platform are needed, so as to satisfy the OEM Systems Communication functionality.

The "On Board Unit" can also exchange data with additional IoT devices such as inertial sensors or the motion sensors of the smartphone: this data are interfaced with the IoT in-vehicle platform using CoAP/6LoWPAN or MQTT protocol as already described, and better implement the concept of "virtual sensors" added to pothole detection.

In order to have a complete vision of architecture of the Italian Pilot Site, external components could also be mentioned. The **AUTOPILOT applications** interface the IoT Platform and implement AUTOPILOT function in the cloud. Each application communicates with the vehicle via the IoT Platform. An application can also comprise a component that runs in the Vehicle Platform. These components can be either an IoT application or an In-vehicle application, depending on the level of integration with the IoT platform.

The **IoT Platform** implements the IoT functions at the Cloud or Edge level. It comprises also other vehicles and roadside elements.

In the Italian use case of Urban Driving, a smart traffic light detects a pedestrian or an obstacle on the lane. The information is processed locally and notified to the RSU using IoT protocols and to vehicles via standard C-ITS messages. Moreover, a connected traffic light sends information about the time to green/red (SPAT/MAP messages).

Figure 58 – Italian PS smart traffic light

The RSU receives the information, fuses the data and sends it by DENM to all the interested actors on the roads. The information from RSUs and OBUs is also sent to the IoT data platform via IoT standard protocols and it can then be processed by the Port Monitoring Centre for real time risk assessment and safety services.

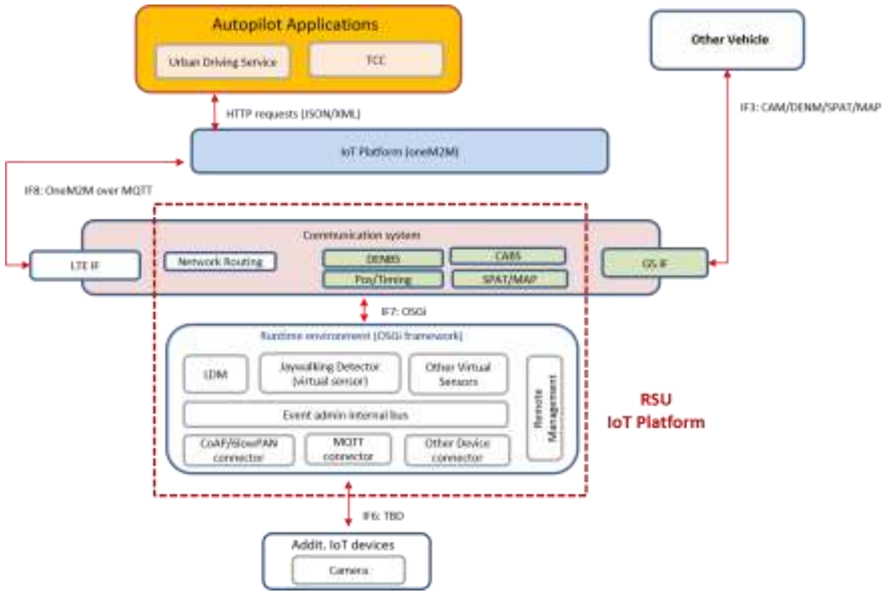

Figure 59 – Road side unit (RSU) software architecture for IoT platform integration (Italian pilot site)

The software architecture scheme showed in Figure 45 is quite similar to the previously explained for the subsection related to the vehicle.
This peculiarity is also given by the modularity and configurability of the designed software, since it is possible to customize it depending on the context in which it is inserted (i.e. in-vehicle IoT platform, Road Side Unit platform). In this case, the Runtime Environment contains a bundle related with pedestrian detection. The module **Jaywalking Detector** represents the implementation of the algorithm that notifies this event when a pedestrian crosses the strip while the traffic light is red. In these conditions, the IoT platform of RSU is interfaced with a camera that may register the wrong crossing of pedestrians and send data to the IoT platform. In this bundle, the data are elaborated and the notification of "detected jaywalking pedestrian" is sent to OneM2M IoT platform exploiting HTTP request (JSON,XML) via OneM2M protocol, or to other vehicles using the CAM/DENM/spat/map interfaces.

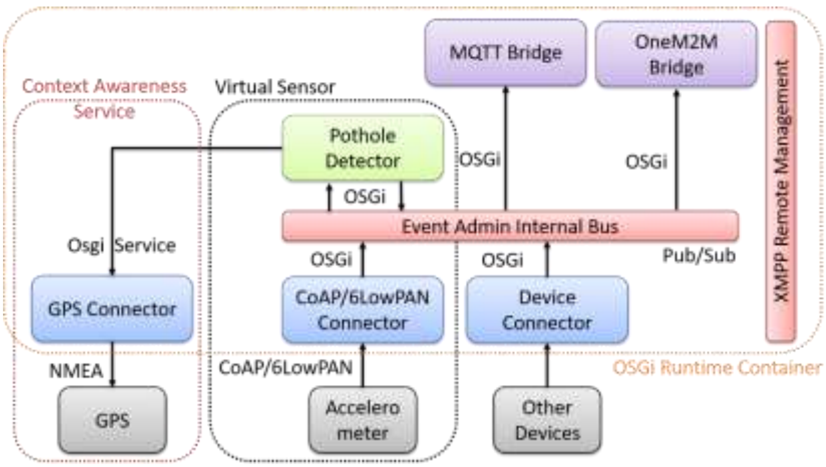

Figure 60 – In vehicle platform, runtime environment (Italian pilot site)

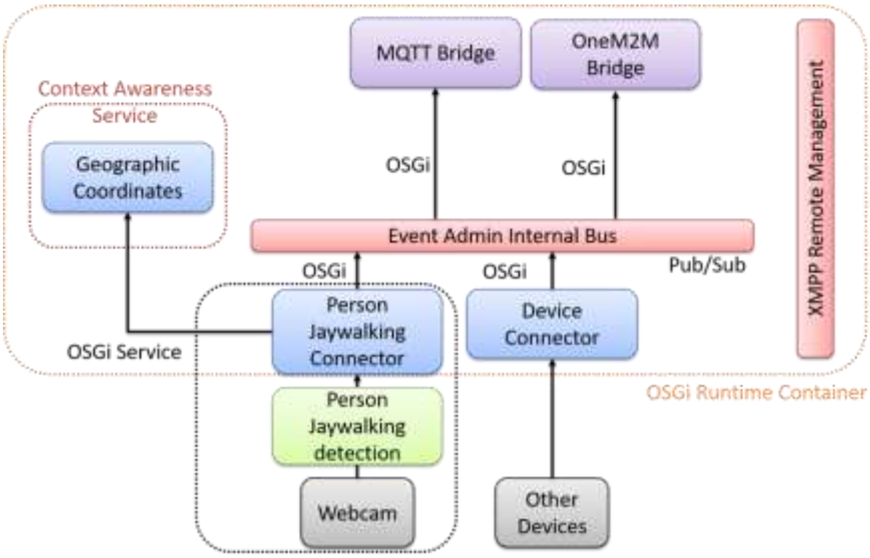

Figure 61 – Road side unit (RSU) IoT platform, runtime environment (Italian pilot site)

Figure 46 and Figure 61 show a zoom on the communication performance between the various bundles of the Runtime Environment in the different scenario: In-vehicle platform and Road Side Unit platform. For each scheme, the main role is played by an internal bus that is responsible for publishing / subscribing the various data sent to the Container Application. As already explained in this chapter, the data can come from both vehicle-based sensors (accelerometer, GPS, webcam) and external components. They are sent via the communication interfaces to the modules (i.e. Person Jaywalking Detection, Pothole Detector) for processing and routing with connectors. MQTT bridge is a connection between the development environment inside the car and interface toward the external world (cloud / edge / other vehicle). This component fulfils part of the APIs functionality and also satisfies the syntactic Interoperability functionality.

As can be seen in the schemes of the chapter, using this type of approach to modify the IoT platform, it is possible to insert new bridge bundles to gain more functionality on the platform itself (at the interface level to the outside/cloud); and/or it is quite easy to sketch in additional bundles connector to interact with new external devices.

NB-IoT puddle detector

The UDP or CoAP messages sent by the NB-IoT puddle detector are provided to the TIM OneM2M platform. The corresponding communication scheme is reported in Figure 62.

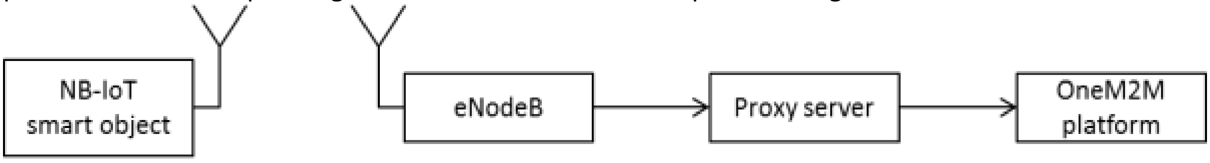

Figure 62 – NB-IoT/OneM2M platform communication scheme

Aim of the proxy server is to adapt the format of the received message to the standard OneM2M, by creating a new JSON message. The main parameters of data transmitted by the NB-IoT module to the OneM2M platform are reported in the Data Model. In addition to the alarm level information, represented by the data field Water_Level_Detector, the sensor position and the time stamp information are also sent. They are respectively represented by Sensor_Position and Sensor_Time_Stamp data fields. For each of the data fields, the Data Model describes the data type, the unit of measurement, the periodicity and the implemented protocol interface, as shown in Table 14.

Table 14 – NB-IoT smart object Data Model

Parameter	Data field 1: Water_Level_Detector	Data field 2: Sensor_Position	Data field 3: Sensor_Time_Stamp
Data type	This field is represented by an Integer [0, 1]. If water is present on the road surface above a fixed threshold, its value is 1, otherwise is 0	This field contains the geographic coordinates of the device, expressed in WGS84 coordinate system. It is constituted by latitude, longitude and altitude, defined as follows: "Latitude": absolute geographical latitude in a WGS84 coordinate system. When information is not available, value will be set to 900000001; "Longitude": absolute geographical longitude in a WGS84 coordinate system. When information is not available, value will be set to 1800000001; "Altitude": altitude in a WGS84 co-ordinate system. When the information is not	Format is yy/mm/dd,hh:mm:ss±zz, where characters respectively indicate year, month, day, hour, minute, second and time zone (the time zone indicates the difference, expressed in quarters of an hour, between the local time and GMT; range is -96 ~ +96). E.g. 6th of May 1994, 22:10:00 GMT+2 hours equals 94/05/06,22:10:00+08

		available, value will be set to 800001	
Unit of measurement	Integer [0,1]	0,1 micro degree	Year, month, day, hour, minute, second, quarters of an hour
Periodic data (yes/no)	Periodic with a period of 1 hour. If a state transition occurs (water present -> not present and vice versa) the first 10 message transmissions have a period of 30 seconds	Periodic with a period of 1 hour. If a state transition occurs (water present -> not present and vice versa) the first 10 message transmissions have a period of 30 seconds	Periodic with a period of 1 hour. If a state transition occurs (water present -> not present and vice versa) the first 10 message transmissions have a period of 30 seconds
Interface from/to OneM2M platform	UDP (JSON payload in hex format) /COAP (JSON payload in hex format)	UDP (JSON payload in hex format) /COAP (JSON payload in hex format)	UDP (JSON payload in hex format) /COAP (JSON payload in hex format)

1.1.65 Integration and communication between 6LoWPAN devices, CNIT IoT-G5 and OneM2M platform

The CNIT IoT-G5 system (shown in Figure 63) is an embedded platform enabling V2X and IoT-based communications by using several wireless technologies.

Figure 63 – CNIT IoT-G5

V2X communications are enabled thanks to 5.9 GHz proximity communications (ITS-G5). IoT features for in-vehicle and sensors communications are based on IPv6 protocol solutions built on top of 2.4 GHz wireless modules developed for constrained devices. Furthermore, IoT-G5 is able to enable LTE communications. More details about its main modules are given below:

- **Vehicular communication module C-ITS**: The solution is based on the Autotalks G5

transceiver (PLUTON), a highly-integrated and automotive-grade V2X RF transceiver optimized for V2X and compliant with IEEE 802.11p (ETSI ITS-G5) standard. Its main features are:
    - Dual-channel 5.18-5.93 GHz, calibration-less operation;
    - Single channel 760 MHz support.

- **GNSS/GPS module**: Geographical positioning is enabled through the Telit Jupiter SL869 module. Such a device is a multi-constellation GNSS module, supporting GPS L1, GLONASS L1 and Galileo E1 constellations;

- **In-vehicle connectivity** provides:
    - CAN bus module supporting both CAN Open and OBD-II interfaces;
    - 2.4 GHz 6LoWPAN module;
    - I/O expender for future applications;
    - Optional Bluetooth 4.0 connectivity.

- **IoT module based on IEEE 802.15.4 technology**: The ATMEL ATZB-RF-233-1-C ZigBit® is embedded as IoT-G5 extender. The module is compatible with robust IEEE 802.15.4 stack that supports a self-healing and self-organizing mesh network, while optimizing network traffic and minimizing power consumption;

- **GSM 4G connectivity**: Long range connectivity is enabled by means of the ME909s-120. The module supports eight bands (B1/B2/B3/B4/B5/B7/B8/B20) in the EMEA region, allowing data rate of 150 Mbps in downlink (Cat 4). Other enhanced features are FOTA, USSD and Huawei at commands software developed for the RSU configuration. It is a flexible and scalable solution for embedded systems, enabling the IoT-G5 acting both as C-ITS station (according to the ETSI C-ITS G5 standard) and border router for Wireless Sensors Networks (WSN) 6LoWPAN compliant. The main RSU functions are:
- Collecting and aggregating data from WSN sensors;
- Communicating with OBUs via CAM and DENM messages;
- Communicating with other information systems (i.e. Traffic Control Centre and OneM2M platform).

In the context of Italian pilot site experimentations, by means of a 6LoWPAN gateway, the IoT-G5 will collect data from a WSN in order to forward it to the OneM2M platform. The corresponding communication scheme is shown in Figure 64.

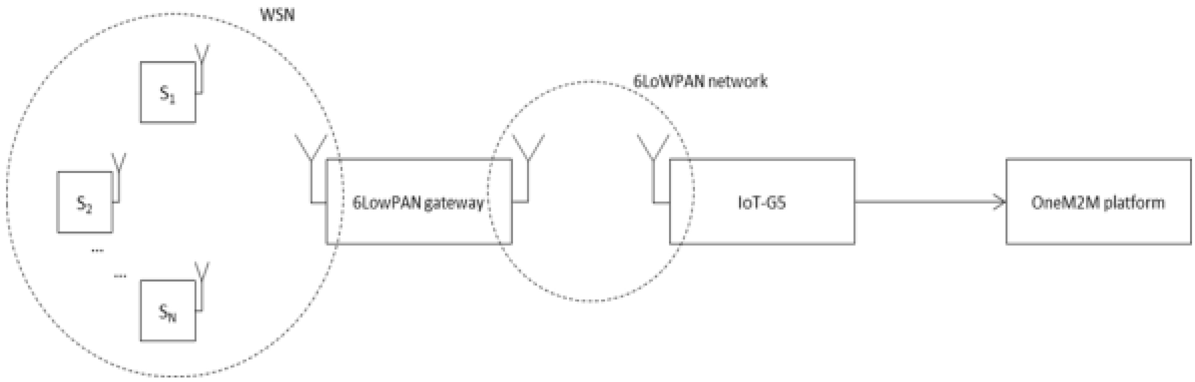

Figure 64 – 6LowPAN/IoT-G5/OneM2M platform communication and integration scheme

The communication between the IoT-G5 and the OneM2M platform is provided by Ethernet or 4G. The main parameters of data transmitted in this context by the IoT-G5 to the OneM2M platform are reported in the Data Model.

In addition to the alarm level information represented by the data field Water_Level_Detector (provided by the puddle detectors according to the measured water level), the sensor position and the time stamp information are also sent. They are respectively represented by Sensor_Position and Sensor_Time_Stamp data fields. For each of the data fields, the Data Model describes the data type, the unit of measurement, the periodicity and the implemented protocol interface, as shown in Table 15.

Table 15 – RSU Data Model for LoRa/6LowPAN network sensors

Parameter	Data field 1: Water_Level_Detector	Data field 2: Sensor_Position	Data field 3: Sensor_Time_Stamp
Data type	This field is represented by an Integer [0, 1]. If water is present on the road surface above a fixed threshold, its value is 1, otherwise is 0	This field contains the geographic coordinates of the device, expressed in WGS84 coordinate system. It is constituted by latitude, longitude and altitude, defined as follows: "Latitude": absolute geographical latitude in a WGS84 coordinate system. When information is not available, value will be set to 900000001; "Longitude": absolute geographical longitude in a WGS84 coordinate system. When information is not available, value will be set to 1800000001; "Altitude": altitude in a WGS84 co-ordinate system. When the information is not available, value will be set to 800001	Format is yy/mm/dd,hh:mm:ss±zz, where characters respectively indicate year, month, day, hour, minute, second and time zone (the time zone indicates the difference, expressed in quarters of an hour, between the local time and GMT; range is -96 ~ +96). E.g. 6th of May 1994, 22:10:00 GMT+2 hours equals 94/05/06,22:10:00+08
Unit of measurement	Integer [0,1]	0,1 micro degree	Year, month, day, hour, minute, second, quarters of an hour
Periodic data (yes/no)	1. If no alarms from sensor network were received during last 24h, 1 keepalive message is sent; 2. If at least 1 alarm	1. If no alarms from sensor network were received during last 24h, 1 keepalive message is sent;	1. If no alarms from sensor network were received during last 24h, 1 keepalive message is sent; 2. If at least 1 alarm

	from sensor network was received during last 24h, a message is sent each hour; 3. In case of sensor state transition: a. Water present: one message is sent each 60 seconds (10 times), then a message is sent each 5 minutes (5 times); b. No water: one message is sent each 60 seconds (10 times), then a message is sent each 15 minutes (3 times).	2. If at least 1 alarm from sensor network was received during last 24h, a message is sent each hour; 3. In case of sensor state transition: a. Water present: one message is sent each 60 seconds (10 times), then a message is sent each 5 minutes (5 times); b. No water: one message is sent each 60 seconds (10 times), then a message is sent each 15 minutes (3 times).	from sensor network was received during last 24h, a message is sent each hour; 3. In case of sensor state transition: a. Water present: one message is sent each 60 seconds (10 times), then a message is sent each 5 minutes (5 times); b. No water: one message is sent each 60 seconds (10 times), then a message is sent each 15 minutes (3 times).
Interface from/to OneM2M platform	CoAP oneM2M compliant (JSON payload)	CoAP oneM2M compliant (JSON payload)	CoAP oneM2M compliant (JSON payload)

As mentioned above, IoT-G5 also communicates with OBUs via CAM and DENM messages. More precisely, in this scenario it implements two operations:

- Collecting CAM messages from vehicles and forwarding them to the OneM2M platform;
- Sending DENM messages both to vehicles and to OneM2M platform.

All these procedures, as shown in Figure 51, are accomplished by using an IEEE 802.11 OBC network and ETSI ITS-G5 protocol stack.

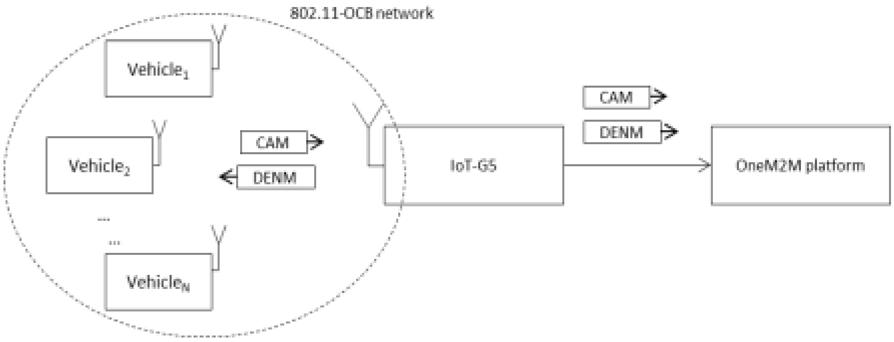

Figure 65 – Vehicles/IoT-G5/OneM2M platform communication and integration scheme.

All details about IoT-G5/OneM2M platform communication are given in the Data Model (see deliverable D2.3).

1.1.66 Security features for IoT devices and IoT communication interfaces

One of the possibilities that will be implemented for assessing security on Italian AUTOPILOT Pilot Site will be a demonstration that will be made with a SIEM device: A Security Information and Event Management. This would be to put probes on the network, maybe on existing machine or on software already available on the network, and to send logs (e.g. syslog) to the syslog engine. A SIEM is generally used to analyse security events. It feeds with data that come from both a traffic analysis and from devices logs. It uses a correlation and interference engine to extract significant events from low-level data piers. It usually displays these events on its configurable dashboard. There are SIEM with inferential and correlations engines of varying complexity (up to SIEM that uses big data and deep learning – although they are probably not completely mature). On the project side, other than host some additional machine, it would have to send the log, preferably with syslog, at runtime, to the SIEM itself. So it would be necessary to configure the devices in order to send the logs with syslog.

The SIEM application will be embedded – or connected – to the AD engine inside the car mainly due to the fact that this engine could take into account the information from a number of security sensors when taking decision about whether or not to completely trust a source of information.

When information that comes from an RSU seems not to be “certified” (unexpected hosts, inconsistent information, etc.) with a security sensor, this information could be either discarded or verified with other sources of information before being accepted.

6.3 Spanish pilot site

The Spanish pilot site has deployed a complex structure where many communication interfaces are involved, as it is shown below.

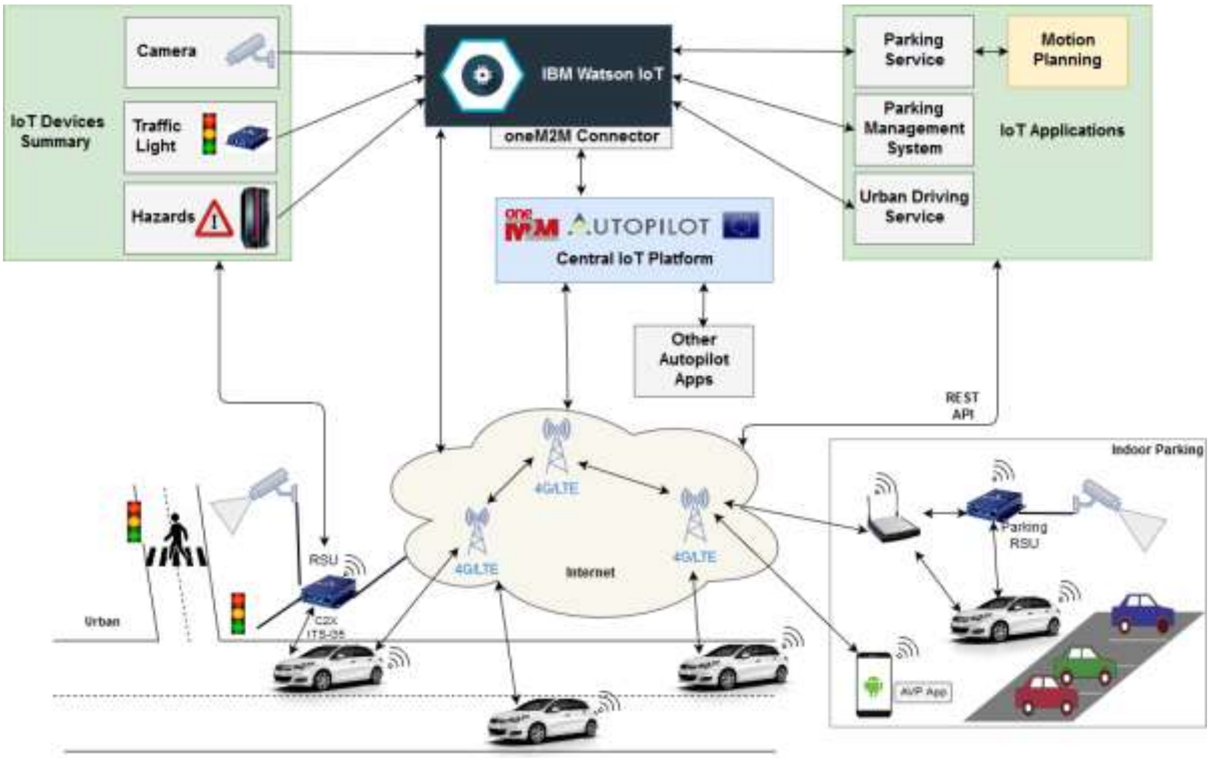

Figure 66: Spanish pilot site communication architecture.

1.1.67 oneM2M communication interface

In the Spanish pilot site, the vehicles integrate an in-vehicle oneM2M IoT platform based on OM2M. This platform directly connects with the oneM2M central IoT platform updating the vehicle's information using the standard messaging format discussed and defined under the T2.3 and the DMAG group.

As both in-vehicle and central IoT platforms are oneM2M, the vehicle-to-central platform communication and the in-vehicle IoT communication are seamlessly forwarded, following both platforms the same messaging format.

This oneM2M communication interface is exclusively used for the in-vehicle IoT communication and to communicate this one with the central IoT platform developed by Sensinov. In order to get the devices information in the oneM2M central IoT platform, the Spanish pilot site relies on the oneM2M-IBM Watson connector provided.

Further information regarding the oneM2M communication interface and the in-vehicle platform in the Spanish pilot site can be found in D2.1 (Vehicle IoT integration report).

1.1.68 IBM Watson IoT devices communication platform

The devices used in the Spanish pilot site are using the IBM IoT Watson communication interface in order to update its information to the IoT platform. IBM Watson provides an API that allows any application or device to communicate with its platform. The devices and applications using this API in the Spanish pilot sites are the traffic control centre, monitoring the traffic lights, the smart camera responsible of detecting VRUs, and the hazards monitoring service, updating any information regarding traffic jams, accidents or road work warnings.

Both the hazards service and the traffic control centre are deployed in servers, so they have direct connection to the network.

On the other hand, the smart camera is connected to an RSU, which is connected to the Internet through cellular network, being the one used to reach the IBM IoT Watson platform.

Further details regarding the IoT devices and its communication interfaces can be found in D2.3.

6.4 Finnish pilot site

Figure 67 shows the communication architecture of the Finnish pilot. For both use cases, the same infrastructure is used: the VTT prototype vehicles and a traffic camera installed on VTT's mobile road side unit. The data of the traffic camera are processed locally and information on objects is transmitted to the IoT platform. Both the vehicle and the mobile road side unit internal network are based on DDS (Data Distribution Network).

Information exchanged between the vehicles, road side unit and the services, like the parking management service, is based on MQTT, and is being sent to an open oneM2M IoT platform. Communication between vehicle, mobile road side unit and the IoT platform uses available mobile commercial network (4G/LTE) or Nokia's Innovation Platform (pre-5G), which is installed in Tampere. Vehicles receive signal phase information from traffic lights through ITS-G5, but alternatively also from the traffic light operator's server (dynniq) over MQTT. Services for parking management and for management of the urban driving use case are developed and integrated with the IoT platform.

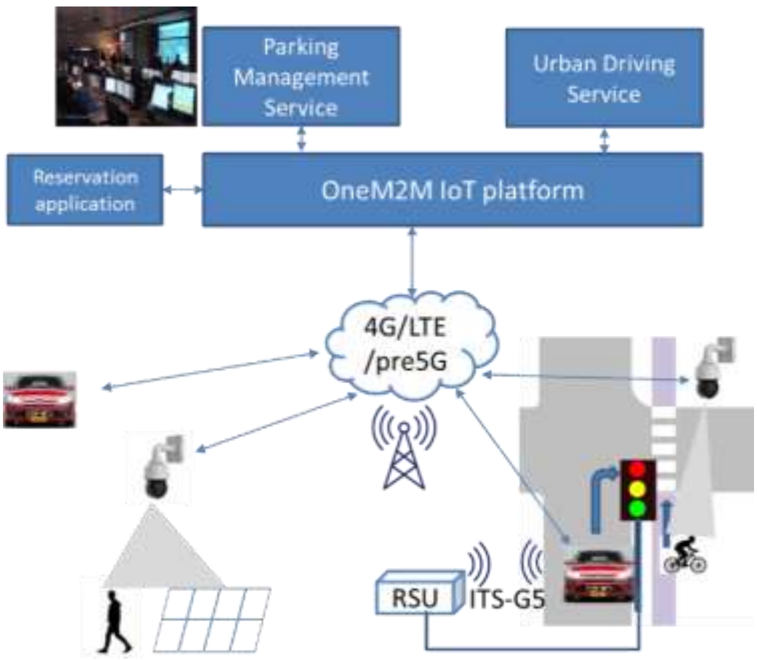

Figure 67. Finnish pilot communication architecture

7. Application integration

Chapter 7 address the high-level integration of the IoT devices and use cases in the different pilot site. The information is linked to how the data acquired through the IoT devices and platforms are integrated in different applications, back ends and cloud services.

7.1 Application Integration - Versailles

For Versailles PS, three main use cases have been identified. As described by Figure 68, the realisation of these use cases involves the development of new IoT services (car sharing service, car rebalancing service, parking slot management, etc.). Applications performing these services are deployed in the cloud. Some of them are offered "as-a-service" to users through their smartphones (for instance, for car sharing service).

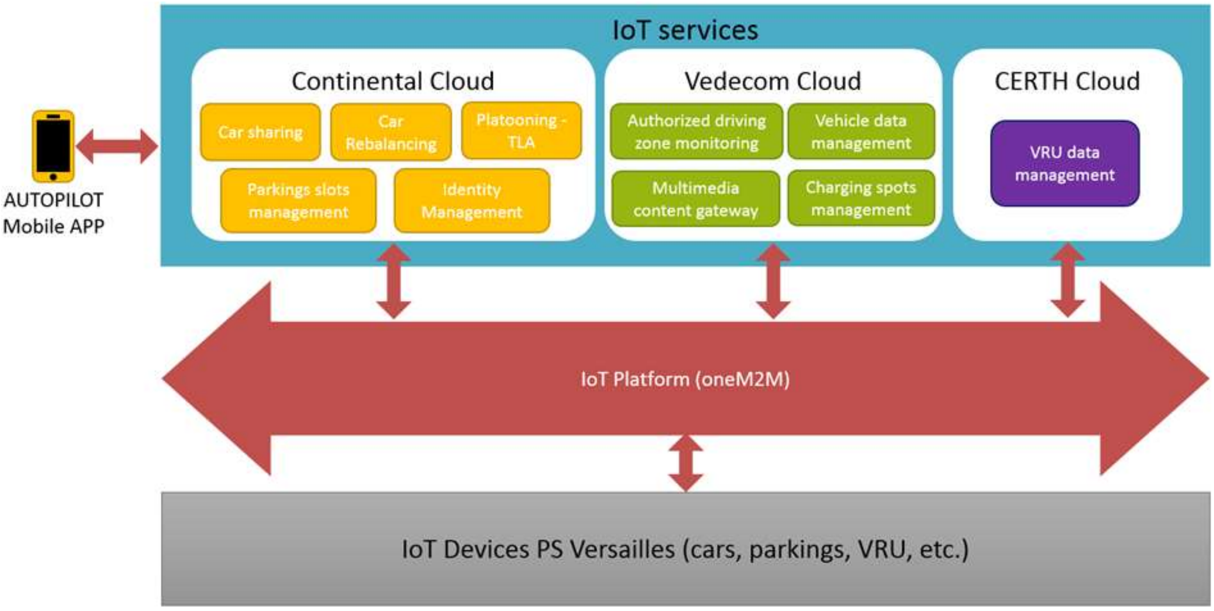

Figure 68 – IoT services for Versailles pilot site

From the point of view of the oneM2M standard, these applications are integrated as AE-IN (Application Entity in Infrastructure Node). They interact through the MCA interface with OneM2M platform in order to retrieve data or to access to IoT devices in a generic way (RESTful exchanges).

Figure 69 shows a zoom of IoT service through there high-level features. They are the order of three main features:

1. **Input data Collection:** these data emanate from IoT devices via OneM2M platform (cars status, parking status, car GPS position, etc.). They can also emanate from AUTOPILOT Mobile App installed in the smartphone of users (e.g. user identity, bookings, etc.). For the realisation of this feature, two phases are identified:
    a. ***Initialization:*** in order to retrieve already stored data and *subscribe* for any new data creation in the OneM2M platform. The subscription requires providing a public endpoint in order to receive *notifications* from OneM2M platform;
    b. ***Data collection*** from OneM2M platform: when new IoT device data are created and/or data/request from users through Mobile application;
2. **IoT Service Handling** for the implementation of IoT Service: It concerns data aggregation, correlation, filtering and processing. For instance, for car sharing service, it can concern filtering available cars according to user request, manage a booking request or manage end

of rental parking spot booking;

3. **Output Data Brokers:** once IoT service handling is completed, it generates output data. These data are sent to the OneM2M platform in order to be available for other IoT services. Depending on the use case, these output data may be sent also to the mobile application (e.g. car sharing use case).

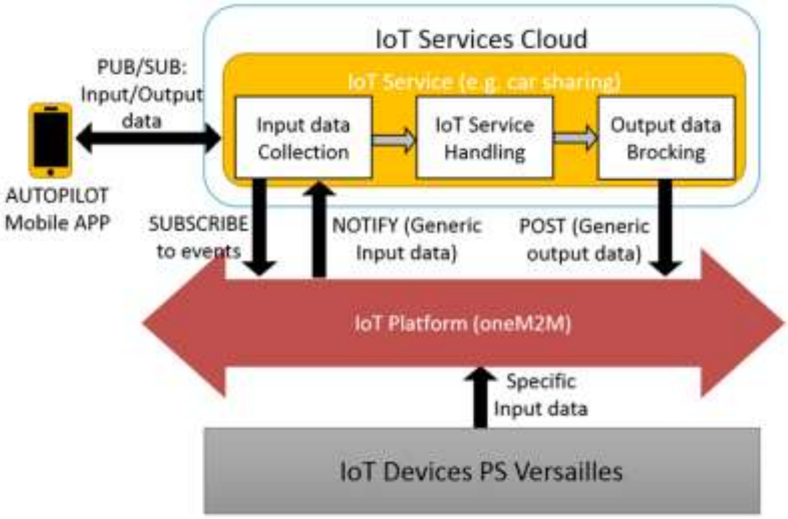

Figure 69 – High-level features of IoT services

7.2 Application integration - Brainport

1.1.69 Automated valet parking use case

The AVP system that has been implemented in the Brainport pilot site supports the vehicle "parking" (from the drop up spot to the parking spot) and "collection" (from the parking spot to the pickup position) scenarios. The AVP system consists of three main parts:

- The AVP applications: provide parking management, user management and routing functions, which are mapped to the corresponding system components, i.e. the parking management service (PMS) App, the user management service and routing service. The application layer provides RESTful interface allowing the AVP mobile App and the parking manager to access the AVP backend services. The PMS has also a bidirectional communication (MQTT interface) with the IoT platform (in the case of Brainport with the Watson IBM IoT platform) to receive all event messages published by the AVP IoT devices and to publish the command messages to the AD vehicles or MAV.
- The IoT platforms: Two IoT platforms are involved by the AVP use case in the Brainport pilot site, the Watson IBM IoT platform and the oneM2M platform. The bidirectional interworking gateway connector allows the interoperability between the two platforms. The TNO vehicle communicates with the OneM2M platform while the DLR AD vehicle with the Watson IoT platform.
- The IoT devices: consist of AD vehicle, road side stationary camera and MAV and the AVP smart phone App. The AD vehicle can receive command message and publish event messages from/to the IoT platform during the AVP "parking" or "collection" process. The MAV and road site stationary camera can publish event message like parking spot occupancy information and obstacle detection to the IoT platform. In addition, the MAV can also receive AVP commands message from the IoT platform after the subscription of these IoT data.

Figure 70 and Figure 71 give a schematic view of the AVP IoT information flow and usages as implemented in the Brainport AVP use case.

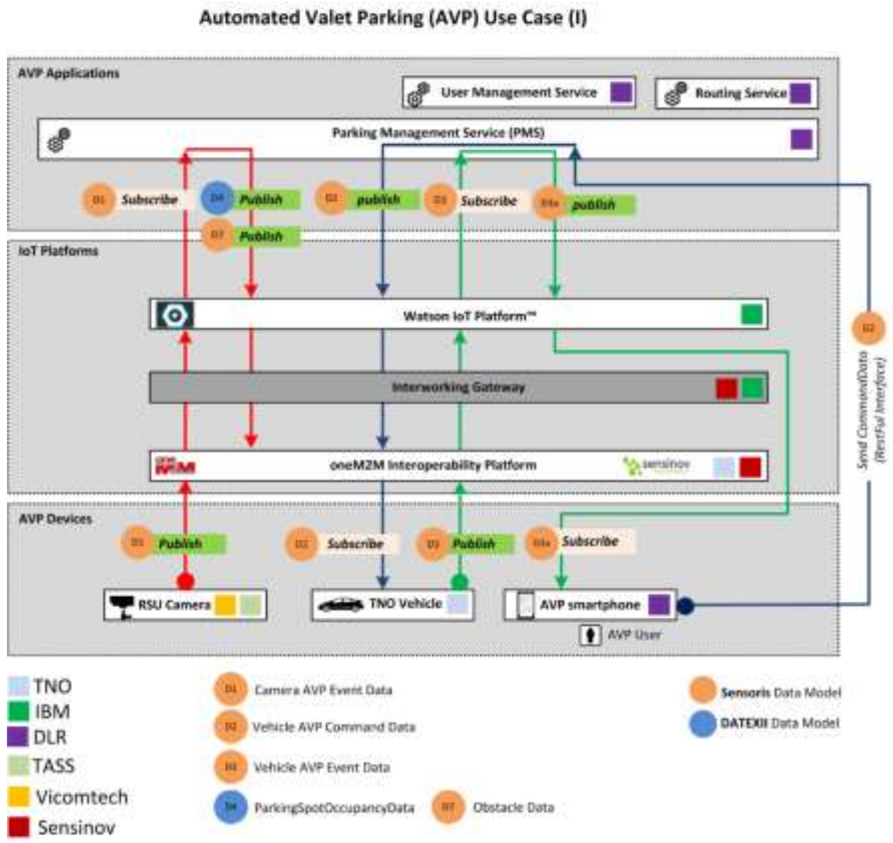

Figure 70 - AVP IoT Information flow and usage Part I

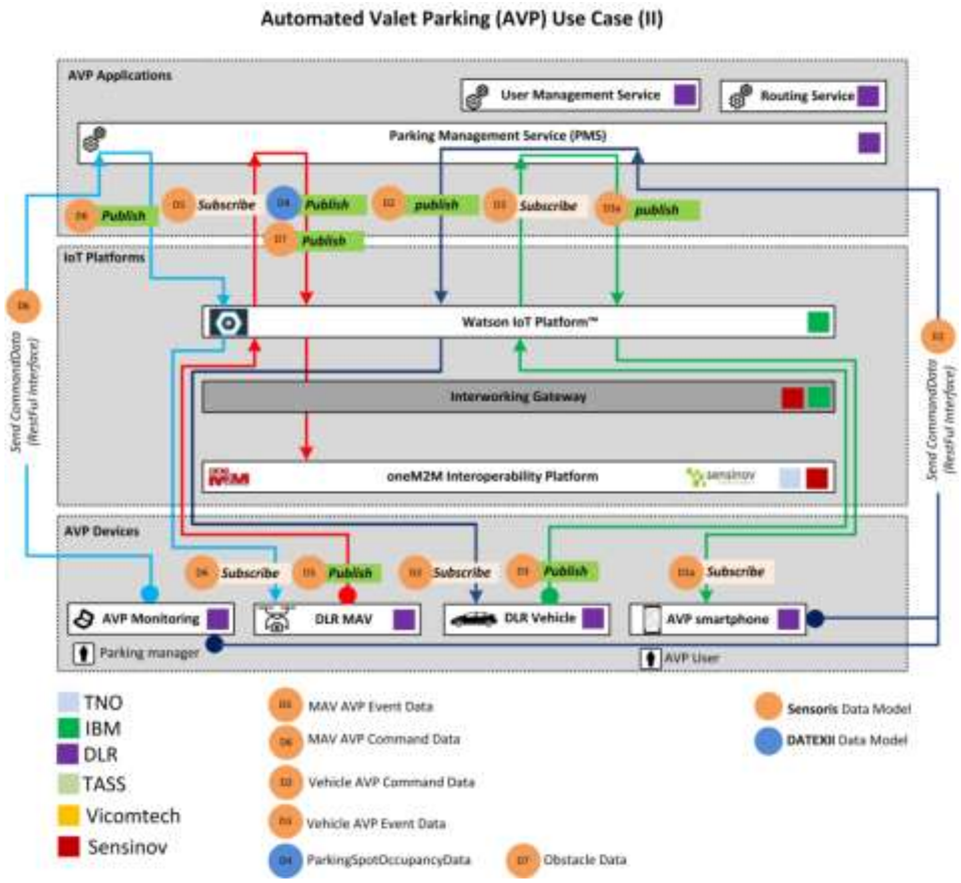

Figure 71 - AVP IoT Information flow and usage (Part II)

The general overview of all IoT data managed by the AVP use case in Brainport pilot site is listed in Table 16.

Table 16 - Overview about the AVP IoT data used in the Brainport pilot site

No.	Identifier	Data	Category	Type	description
1	D1	Camera Event Data	Camera Data	- Parking spot occupancy detection (raw data) - Obstacle detection	The camera can publish the data into the IoT platform and the AVP PMS backend Application can subscribe to the IoT platform to receive the data
2	D2	Vehicle AVP Command Data	User Data	- AVP Route (drop off-position to parking spot or parking spot to pick up position) - AVP Instruction (vehicle parking, vehicle collection, vehicle charging, vehicle re-parking)	The AVP user via his AVP mobile app sends the command to the AVP PMS application; the application publishes the command to the IoT platform. While the vehicle subscribes for this data can receive the command from the IoT platform
3	D3	Vehicle Event Data	Vehicle Data	- AVP Status Message - Vehicle dynamics data - Vehicle position data - Parking spot detection - Vehicle meta data - Etc…	The vehicle publishes the event data into the IoT platform. While the AVP PMS application subscribes by the IoT platform can receive the vehicle event data

4	D4	Parking Spot Occupancy Data (processed)	Parking data	- The processed parking spot occupancy data	The parking data provided by the roadside unit camera and MAV can be processed by the AVP PMS application and the processed parking spot occupancy information can be published to IoT platform as ParkingStatusPublication using DATEX II data model
5	D5	MAV AVP Command Data	User Data	- AVP waypoints, Route, AVP Area to fly - List of Parking spots to check - AVP Instructions	The AVP Parking Manager via the AVP monitoring tool sends the command to the AVP PMS application; the application publishes the command to the IoT platform. While the MAV subscribes for this data can receive the command from the IoT platform
6	D6	MAV AVP Event Data	MAV Data	- AVP Status Message - MAV meta data - MAV dynamics data - MAV position data - Parking spot detection (raw data) - Obstacle detection (raw data) - …	The MAV publishes the event data into the IoT platform. While the AVP PMS application subscribes by the IoT platform can receive the MAV event data
7	D7	Obstacle Data (processed)	Obstacle Data	- The processed obstacle data obtained by Road site unit camera or MAV	The obstacle data published by the road side camera and MAV into the IoT platform will be processed by the AVP PMS application. The AVP PMS application after processing publishes the obstacle data into the IoT platform and is available for other IoT applications

1.1.70 Car sharing use case

IBM IE developed a ridesharing service (from now on referred to as RaaS) and designed its architecture. A ridesharing service is a service that given a set of customer requests and a fleet of intelligent vehicles, determines the best match in terms of which vehicle will serve which customers. In (real-time) ridesharing, the customer requests are revealed over time and the service changes the already determined schedule to accommodate new customers. In ridesharing, multiple customers can be serviced by the same vehicle.

An architecture diagram is shown on Figure 72 for this use case. The implementation assumes that the pilot site vehicles are connected directly to Watson IoT platform.

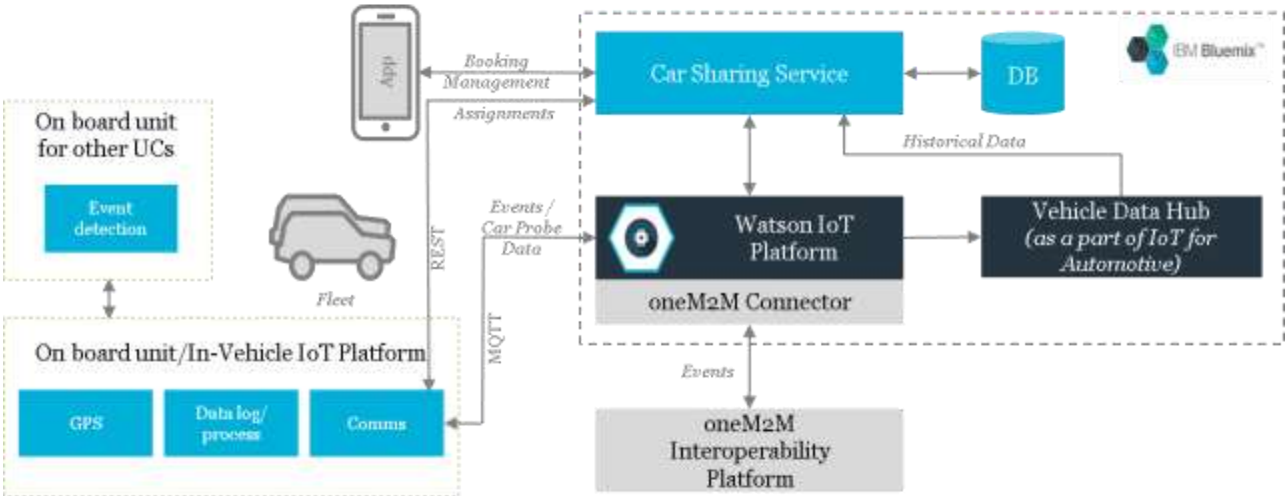

Figure 72 – Architecture diagram for the car sharing use case

There are two IoT components of the central IoT platform shown on the figure:

- A Watson IoT Platform that is a pub/sub broker supporting the MQTT protocol for publishing and subscribing to device data. Connected devices publish data using events. Devices control the content of the events and assign a name for each event that is being sent. When an event is received by the Watson IoT Platform, the credentials of the connection on which the event was received are used to determine the device from which the event was sent.
- The Vehicle Data Hub (VDH on the diagram) is a front-end interface component of IoT for Automotive that retrieves and processes data from the connected devices. In AUTOPILOT, the main source of data is Watson IoT platform. When Watson IoT receives a new message from a connected device and this message is a car probe message, this message is passed to VDH. When data is received by the VDH from connected vehicles, the data is normalized and then integrated with asset information, such as vehicle or driver data. From the VDH, the vehicle data is then dispatched to other components connected to IoT for Automotive.

The car sharing application connected to the central IoT platform processes ride requests in the following way:

- The application receives ride requests from customers via a front end (either web-app, or smart phone app, provided by Gemalto). A ride request is a tuple containing pick-up point, destination, time windows and if the rider wants to share one's ride or ride alone.
- The application is connected to a database that stores the fleet information (e.g. location, status);
- The requests are batched together any 10-30 seconds and the application's optimisation engine computes the best vehicle to customer match;
- The application then informs the vehicles either via the IoT platform or via preferred communication on the decided assignments, destinations, routes and schedule;
- The in-car IoT platform displays the directions, routes and schedule to the driver;
- The application informs also the customers and keeps them up-to-date with their schedule;
- The vehicles are always connected to the IoT platform to send their GPS and status data, as well as possible events. These data are used to update the database that the application accesses to provide up-to-date schedules and directions.

1.1.71 Urban driving / Car rebalancing use case

Different applications are implemented in this use case: OneM2M has been chosen to be the main central IoT platform to which FIWARE and WATSON are interconnected. HUAWEI OceanConnect is separately connected to the IoT devices (smartphone and vehicle). See Figure 73.

The smartphone pushes data to OneM2M, which the Rebalancing service on WATSON uses. CEMA directly connects to the Wi-Fi Crowd Detector to estimate the crowdedness with the CEMA application running on FIWARE. The output of FIWARE is published to OneM2M, to which the vehicle is connected in order to retrieve the CEMA data.

HUAWEI GeoFetching retrieves data from the smartphone directly and together with the location of the vehicle, it publishes back to the vehicle any data from smartphones (i.e. Locations) in the near area of the vehicle.

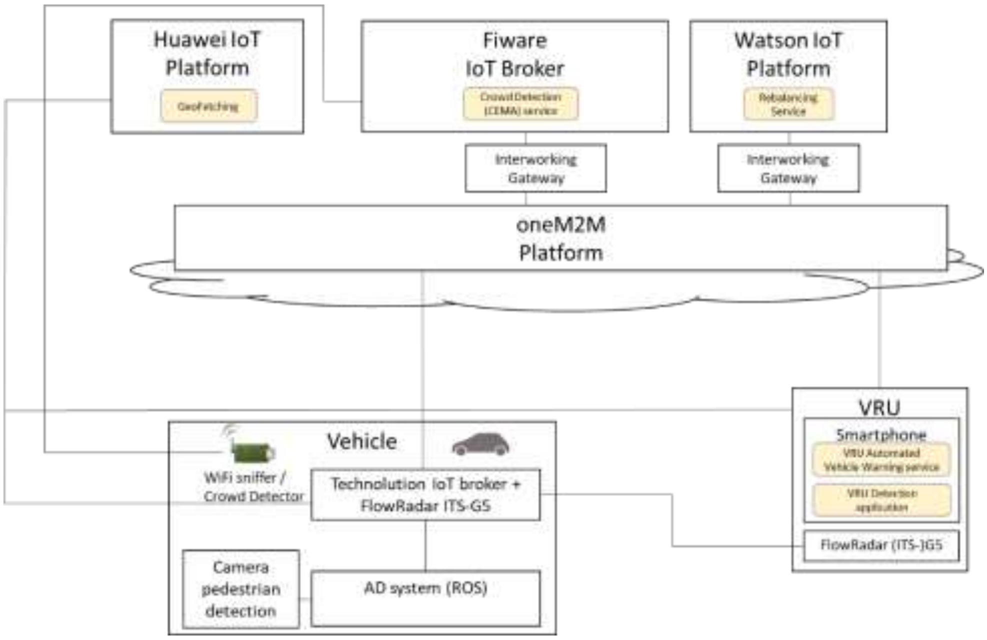

Figure 73 – IoT device and IoT platform integration diagram for Brainport Urban Driving / Car Rebalancing use case

1.1.72 Platooning use case

In the same style, the Platooning use case implementation has been depicted below. The Platooning use case rests on various services: Platooning Service (TNO), Traffic Manager Service (TASS), Traffic Detection service (TASS) and the LDM service (TNO, under construction). The core platform is the oneM2M platform. Besides the Infrastructure node, a middle node instantiation plays a role to support the gathering and use of local dynamic map data. The key service is the Platoon Service, which guides our project vehicles during platoon formation and actual platooning. It relies on the availability of real-time vehicle registrations (N270) and traffic light data, which are both pushed to the OneM2M platform. It also relies on the Traffic Manager application (TASS) and we anticipate a coupling between the Platoon Service and the car sharing service (IBM) to facilitate the matching of vehicles that want to platoon. The LDM service, which involves the participation of other vehicles (consent), provides a shared world model that can be used by the subscribed vehicles (which would include the platooning vehicles). The other vehicles (external to the project) will be involved through an app carried by the users of these vehicles.

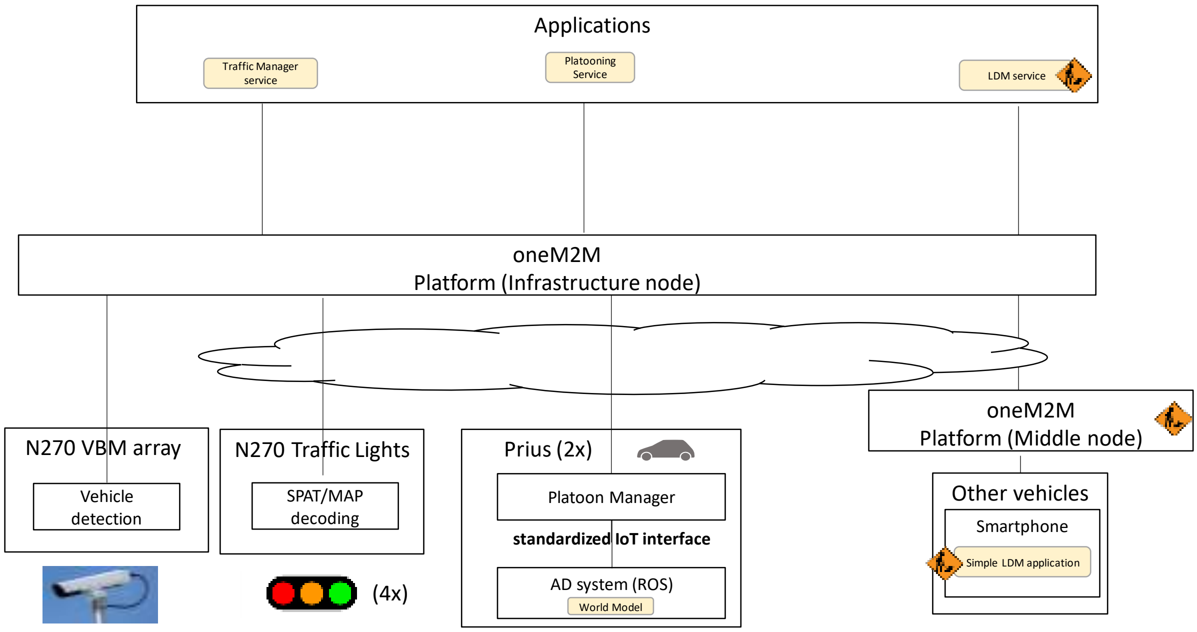

Figure 74 - IoT device and IoT platform integration diagram for Brainport Platooning use case

1.1.73 Highway Pilot use case

Three core cloud applications are integrated through the OneM2M IoT Platform, exchanging 3 classes of messages (ANOMALY, HAZARD, ADASIN):

- Hazards Learning application in Valeo Cloud
- Live Maps application in TomTom servers
- Traffic Management application by TASS for Control Centre

The Hazards Learning application collects all anomalies detected, whether they are originated from vehicles of from roadside cameras. The application compares their characteristics (geographical, confidence, etc.) in order to aggregate them into knowledgeable hazards. These are then published for other applications to use. The Traffic Management application by TASS subscribes to hazard containers in the OneM2M IoT platform.

Whenever new hazards are published, an application server at TASS gets the notification which then hand over them to the traffic management application to display them in a map as shown below. If the hazard is on A270 roads in the field of view of TASS roadside camera, the traffic operator can visually check the hazard and process them.

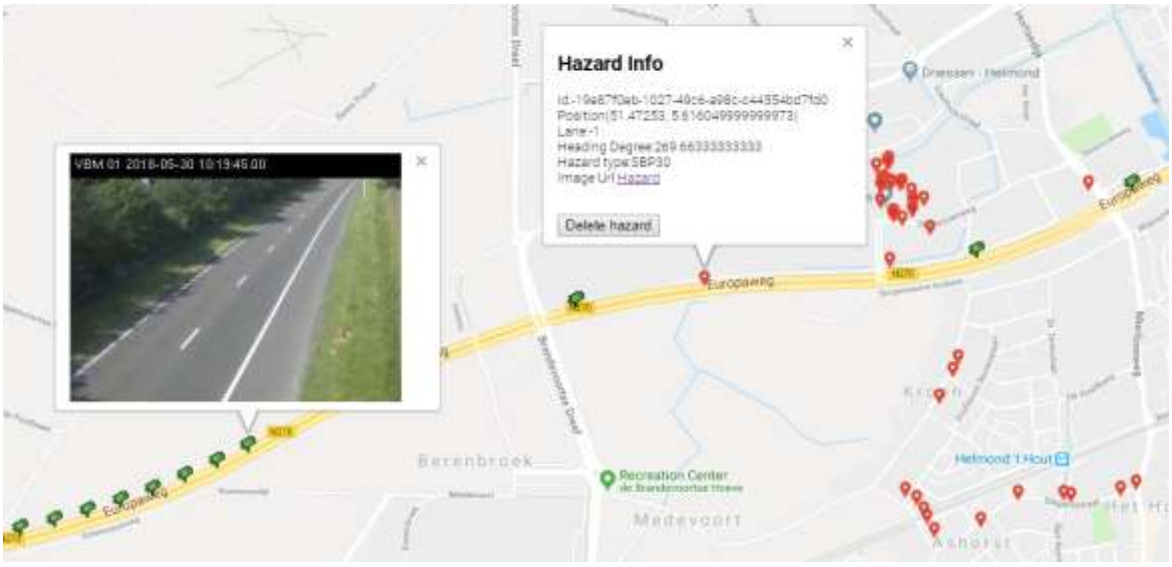

Figure 75 - Hazard Information in Google Maps

The traffic operator then sends the ADAS instruction corresponding to the hazard received with the help of the GUI. This includes recommended speed, time interval, lanes to avoid, etc. The ADAS instructions data model is currently being standardized (for AUTOPILOT community) in the data modelling group. The ADAS instructions will be sent to the ADAS containers in the OneM2M IoT platform. The vehicles subscribed to this container will get this information and adapts their driving accordingly. The road side cameras also detect anomalies in the road using the anomaly detection algorithm provided by Vicomtech. These algorithms are running in the TASS servers and use Advanced Message Queuing Protocol to communicate with TASS OneM2M Interface application, which then publishes the anomalies to the Anomaly container in the OneM2M platform. Valeo application subscribed to this information will get these anomalies and issues hazards after processing them.

The Live Maps application collects Hazards and ADAS Instructions into mapping system. From then on, whenever a vehicle enters an area with relevant objects, the vehicle will receive them. This is enabled by the integration into the car of a location-aware client that continuously checks and syncs.

7.3 Application integration – Livorno-Florence

The concept of “Virtual Sensor” is part of Italian IoT in-vehicle platform concept and is therefore bound to use cases. With the help of Fog computing, object virtualization aids to address the issues of heterogeneity, interoperability, multitenancy, scalability, counter-productivity, mobility and protocol inconsistency that commonly exist in IoT.

Fog computing essentially extends cloud computing and services to the edge of the network, bringing the advantages and power of the cloud closer to where data is created and acted upon. The goal of fogging is to improve efficiency and reduce the amount of data transported to the cloud for processing, analysis and storage. This is often done to improve efficiency, though it may also be used for security and compliance reasons. In a fog environment, the processing takes place in a data hub on a smart device, or in a smart router or gateway, thus reducing the amount of data sent to the cloud. Fog computing reduces the bandwidth needed and reduces the back-and-forth communication between sensors and the cloud, which can negatively affect IoT performance.

In relation with the concept of Fog Computing, since there is no sensor on the vehicle that "physically" detects the roadway potholes, it is thought to use a combination of sensors that can already be integrated into the OEM dispositive (e.g. accelerometers, gyroscopes, etc.). Alternatively sensors from external devices can be used to be placed on-board e.g. smartphones, cameras, external accelerometers, etc., to recover the same type of data (acceleration, orientation, etc.). In this way, data from different devices are fused together and processed: from these sensor fusion outputs, the data that can be used to detect the road holes. The result of this fusion is therefore a "pothole detector" and these elaborations are sent to the cloud OneM2M platform or RSU or other vehicles (via communication system).

In the Italian Site Tests, the goal is to show how the combined use of IoT and C-ITS can mitigate the risk of accident for an AD car, when at a certain point the road becomes dangerous because the road is wet and/or rough. The IoT sensing infrastructure on the highway contributes to improving safety and easing traffic management.

IoT sensors placed along the highway monitor continuously the presence of puddles. When the hazard is detected, the sensors send an alert to the RSU with detailed information, using IoT standard protocols. RSU broadcasts the info to vehicles (DENM) and to the Traffic Control Centre (TCC). The TCC validates the alert and forwards the DENM message to farther away RSUs. At the same time, the TCC feeds the ETSI OneM2M platform with alert-related data. Then the information is consumed by the Connected eHorizon (CeH) application from CONTINENTAL, transmitted to FCA cloud as a modified dynamic speed limit that considers the generated dynamic event. FCA cloud immediately notifies to enabled vehicles the updated information for CeH devices installed on prototypes. The in-vehicle application feeds the appropriate autonomous functions that perform the necessary adaptation of the driving style in a "smooth" way in combination with information obtained from DENM. A notification/warning through the in-vehicle HMI can be generated.

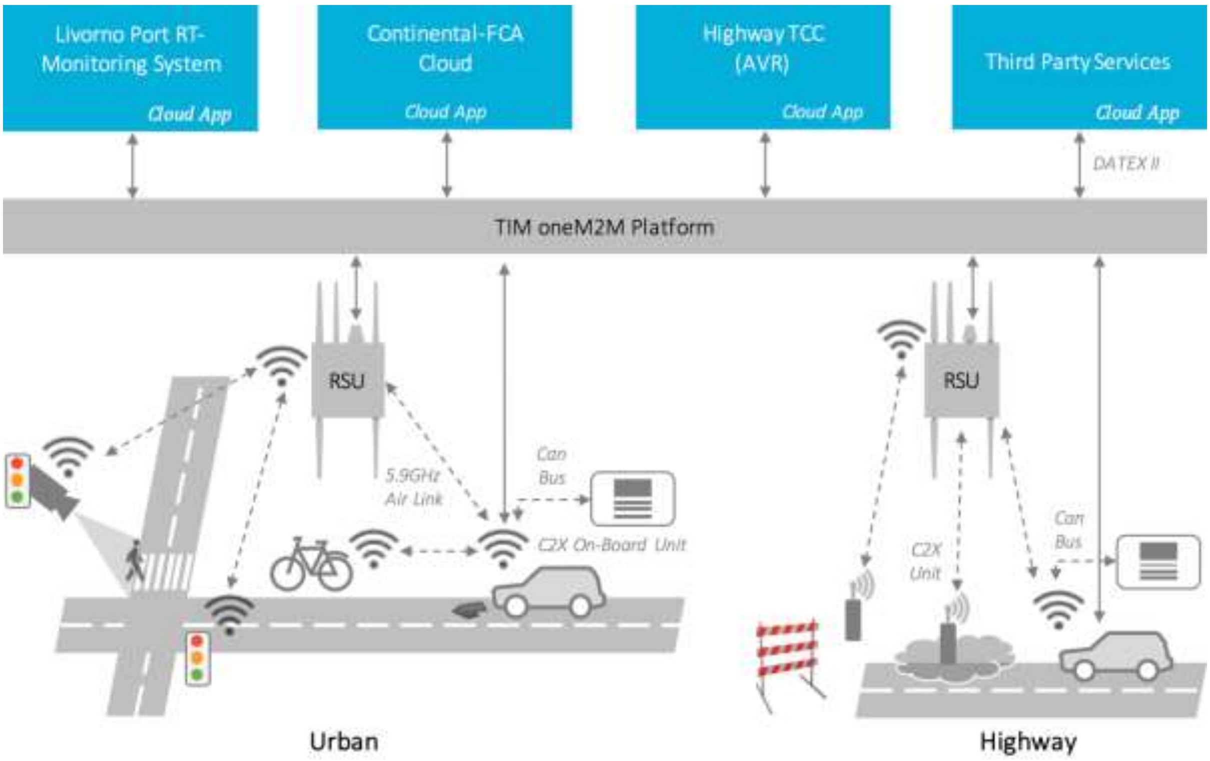

Figure 76 – Livorno Pilot Site Overall Architecture

Combination of "Long range" information provided by IoT and related cloud, and "Short range", information provided by ITS-G5 notification is expected to enhance capability of an AD vehicle to perform manoeuvres with a more relaxed response time.

Also, in the urban driving use case, the interaction with fleet management service for adapting the driving style to the situation enhances safety and eases traffic management.

The ITS ecosystem is populated by new sources/consumers of information, namely AD cars, road side smart objects, sensors that contribute to improve services and travel experience.

As described by Figure 76, the realization of these use cases involves the development of new IoT services (highway service, urban driving service, TCC, etc.). Applications performing these services are deployed in the cloud.

AD deployment is accelerated by adding redundancy to VRU detection (both vehicle and infrastructure are detecting AD function) and is enhanced in the preventive safety area, because IoT anticipates possible threats, especially in relation to the current traffic light phase, and the vehicle is expected to tune its speed accordingly.

AD deployment is accelerated by IoT, because dynamic speed limit functionality can be done directly by the Connected E-horizon via the One M2M platform interface in the cloud (i.e. outside the vehicle). This is a complementary approach with respect to the in-vehicle data fusion between speed limits from V2X and E-Horizon: it can give redundant data if both ways are implemented, but it can also enable IoT dynamic speed limits just with the E-horizon.

AD deployment is enhanced by IoT hazard detection because hazards on the road can be anticipated through infrastructure-to-vehicle communication. The IoT/V2X value is to extend the field of view, and trigger smooth control manoeuvres, while the on-board sensors play a stronger role in the proximity of the danger.

7.4 Application integration – Vigo

1.1.74 Urban driving use case

For the urban use case, the Spanish pilot site is using three different IoT platforms, which are integrated and interconnected.

The different devices: smart camera, traffic control centre and hazards service are connected to IBM IoT Watson; the vehicle has its own in-vehicle oneM2M platform based on OM2M and is also connected to the central IoT oneM2M platform.

The devices are constantly updating their status on the IBM IoT Watson platform. The pilot urban service is subscribed to these status updates and can provide the information to the connected vehicle, which will publish the relevant data into the in-vehicle platform. Finally, the vehicle will also publish its relevant status to the central IoT oneM2M platform.

The integration of these platforms and devices will allow the connected vehicle to receive and send all the needed information that enable and potentially improve its autonomous driving functionalities.

1.1.75 Automated valet parking use case

IBM RE implemented a parking spot service and developed its cloud architecture. The service allows to collect data about parking spots from different IoT sensors and provides full access to this data and its derivatives through the public API.

Parking information might be provided, for example, by a parking garage and include static and dynamic parts. The first part contains information like drop off and pick up locations, parking fees and opening times. The dynamic part includes real-time information about parking spot occupation. The service stores this information and allows getting latest occupancy states, historical retrospective and forecasts of parking spot occupation. All these features are accessible via the RESTful API.

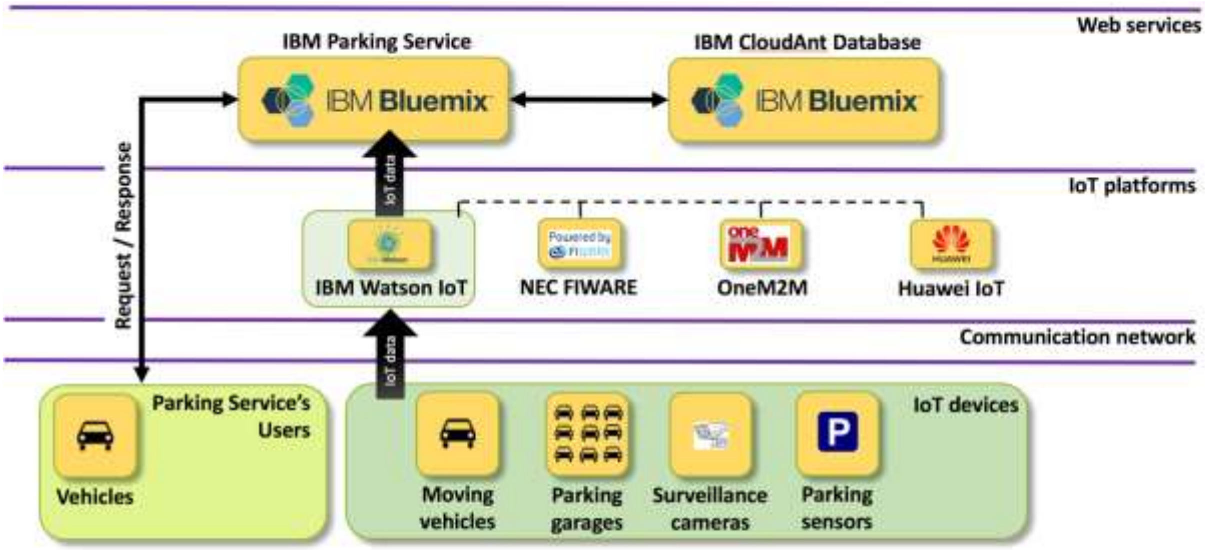

Figure 77 – Architecture diagram for the parking service use case

An architecture diagram is shown on the Figure 77 – Architecture diagram for the parking service use case that is agnostic to the specific pilot sites and might be used in different places.

The IoT platform used by the parking service is a Watson IoT Platform that supports publishers and subscribers to the MQTT messages coming from IoT devices. Connected parking sensors like parking garages or surveillance cameras publish their data using standardized parking messages. Each message contains unique identifiers of the parking spots or their groups with current occupancy states.

The parking spot application connected to the central IoT platform process requests about parking spot availability with the following steps:

- The application receives a request from the client that contains a geographical location with a radius;
- The application requests identifiers of existing parking spots given area from a database that stores a map with available parking spots;
- The application requests a current occupancy for each parking spot selected on the previous step;
- The application sends a response to the client that contains information about all parking spots;
- The application always operates with actual information about parking spots coming from IoT devices.

7.5 Application integration – Tampere

Two use cases are identified in the Tampere pilot site: Automated Valet Parking and urban driving. The architecture of the Finnish pilot is shown in Figure 67.

For Automated Valet Parking, a parking management application is being developed. This application has the following roles:

- Assignment of parking place and route to the vehicle. For the parking place assignment, a reservation is made through the Mattersoft application, which is developed in the H2020 Transforming Transport. Communication between the IoT platform and the Mattersoft application is via RESTful API;

- The camera images are processed in the mobile road side unit of VTT. Through the use of neural network technologies, the software assesses if the parking places are free and if there are objects on the possible vehicle paths. This information is sent to the IoT platform over MQTT;
- The application assigns a route based on the assigned parking place and potential objects on the route;
- The application gives the permission to start the vehicle to drive to the parking place;
- Monitoring of the parking manoeuvre during unmanned driving. The system will potentially monitor through video (from the traffic camera) if the manoeuvre is performed safely.

For urban driving, the information of the traffic camera is processed locally in the mobile road side unit and forwarded to the IoT platform. The urban driving service forwards both the traffic light status and the information on pedestrian presence from the traffic camera to the vehicle.

8. Verification, validation and testing for selected cases

The verification, validation and testing (VV&T) exploits parts of the work results carried out in deliverable D5.3 (Performance and KPIs for autonomous vehicles and IoT pilot impact measurement).

Chapter 8 describes the VV&T of selected cases for IoT devices used in autonomous vehicles use cases for achieving acceptable levels of safety and assurance for the autonomous vehicle applications. These are VV&T methods used at the IoT devices, sub-systems, communication and integration into IoT platforms.

The software, sensing and connectivity requirements of IoT for autonomous vehicles devices require proper methods and concepts for VV&T. Different approaches include technology-specific, traditional automated equipment that uses different hardware to conduct specific VV&T, software-centric, etc. Testing is performed to verify that IoT devices perform as expected and are within specifications and requirements for the IoT autonomous vehicle use case. VV&T confirms the ability of the IoT devices used in autonomous vehicle use cases to meet industry standards, such as interoperability, interfaces, protocols, etc.

The VV&T of IoT devices used in autonomous vehicle use cases are part of the activities addressing the trustworthiness of these systems by considering the technical features of the IoT devices and achieve the required safety, reliability, availability, resilience and security.

Few elements to be checked that were considered:

- EMI/EMC: Electromagnetic interference (EMI) for the IoT devices operating simultaneously in close proximity to one another or using different frequencies and protocols;
- Wireless connectivity of autonomous vehicle safety-critical IoT devices performing in environments with multiple users, with different wireless technologies, in the same spectrum. VV&T of the load and robust wireless connectivity;
- Co-existence and interference: VV&T of autonomous vehicle safety-critical IoT devices for interference between devices;
- Connectivity network readiness: VV&T of autonomous vehicle safety-critical IoT devices supporting a range of wireless communications technologies, and the capacity of the infrastructure networks to support these technologies (i.e. different environments and locations, RF conditions that differ significantly, etc.);
- Safety-critical IoT device maturity and security: Type of performance tests and conformance verification.
- Service quality and performance considering the updates and upgrades to network level and VV&T of autonomous vehicle safety-critical IoT devices.

8.1 Integration of devices/applications to oneM2M IoT Platform

To show the big picture in terms of development and integration, we are summarising all the interworking components required for IoT platform integration among the pilot sites based on the work carried out in task T2.4.

The oneM2M standard defines two mechanisms to connect oneM2M and non-oneM2M devices/applications into the IoT platform:

- Native oneM2M devices/applications: Can interact directly with the oneM2M platform using the MCA interface;
- Non-oneM2M devices/applications: A dedicated Interworking Proxy Entity (IPE) shall be

developed and deployed for this purpose. The IPE provides interworking between oneM2M platform and specific IoT device/application technologies or protocols.

The two mechanisms are illustrated in Figure 78.

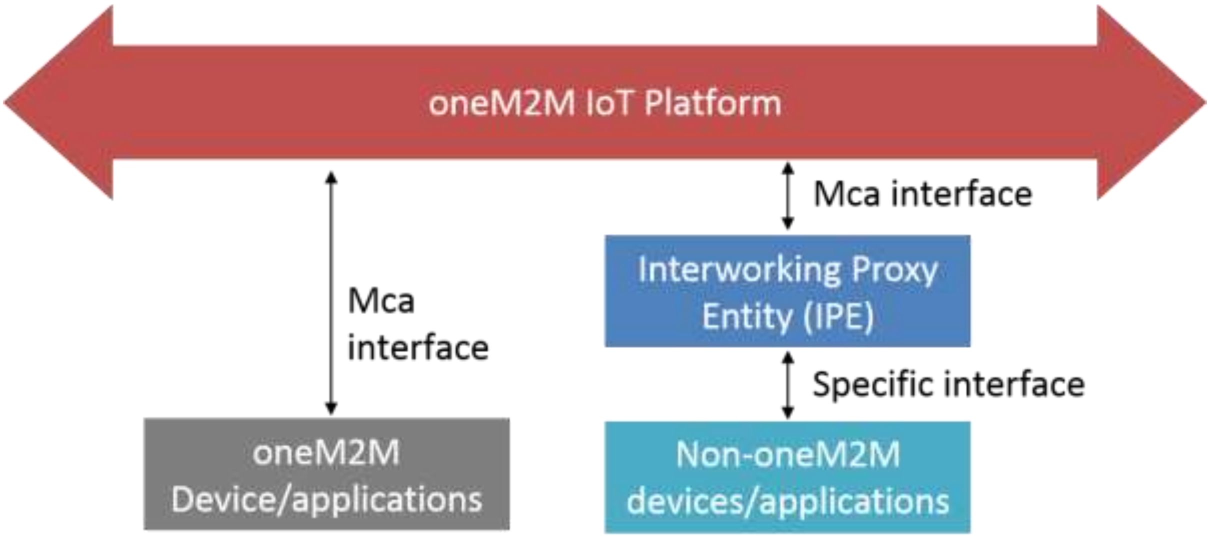

Figure 78 – Mechanisms of connecting oneM2M and Non-oneM2M devices/applications into the IoT platform

Required interworking proxies' entities per pilot site

A set of interworking proxies' entities (IPEs) are identified and shall be developed to integrate pilot site devices/applications to the oneM2M IoT Platform. Some IPEs are duplicated between pilot sites, e.g. CAN, CAM, DENM, etc. In some cases, vendor-specific devices will communicate directly with specific IoT platforms using proprietary protocols. The required IPEs are illustrated in Figure 79.

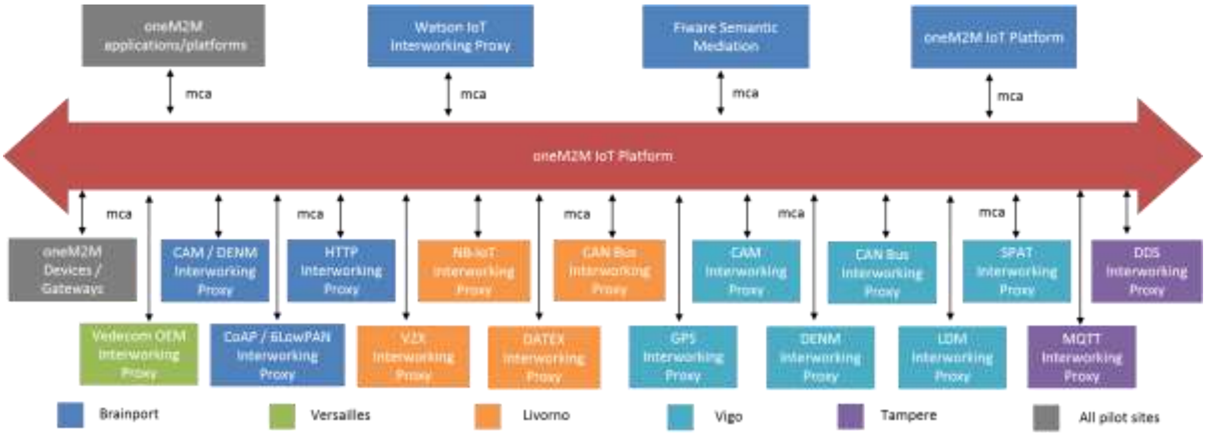

Figure 79 – Required IPE per pilot site overview

8.2 Smartphone connection and testing with OneM2M and HUAWEI OceanConnect

MQTT connection (Smartphone to OneM2M platform):

A connection with the over-arching OneM2M-platform is created via the MQTT protocol. In the smart phone application, a plugin has been used for MQTT protocol connections called Paho MQTT client. With these plugin connections to a broker, publishing of messages to this broker as well as subscribing to topics on this broker has been handled. As discussed before MQTT uses Topic-Based routing. After connecting to the broker with credentials, messages can be published to certain topics. These topics can be made up out of four layers:

- Common Services Entity (CSE)
  - Application Entities (AE)
  - Containers (cnt)
    - Content instances (cin)

The Topic-Based routing is the Topic routing that is used in the message format to route to different containers on a specific AE. However, the broker also has a certain topic which the message has to be sent to. This is a unique topic that comes with the credentials that are defined in the JSON message that is sent. ”/server/rqp/*AE- ID*/server/json” for publishing and ”/server/rsp/server/*AE-ID*/json” for subscribing to a certain AE, for which, as a user, you have the credentials.

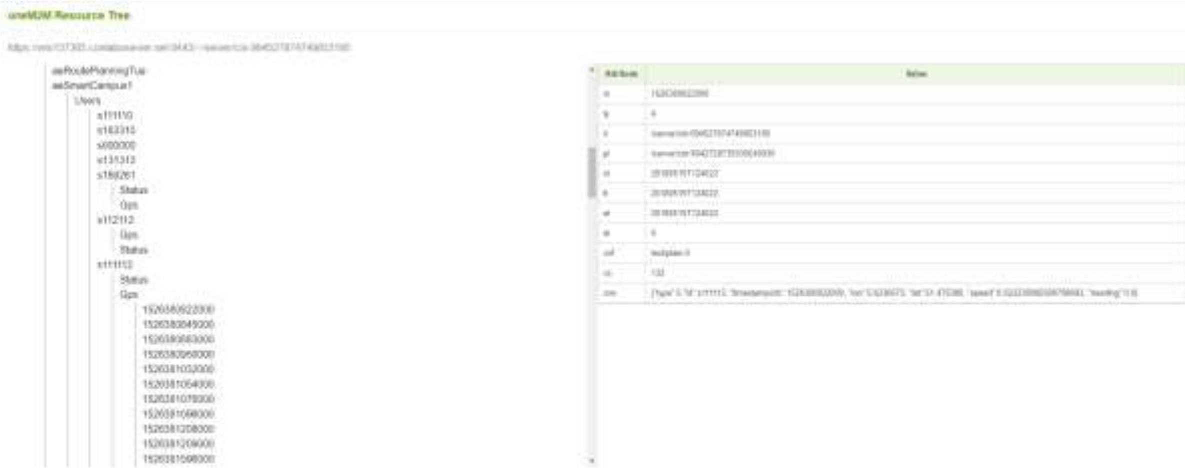

Figure 80 – The SmartCampus structure on OneM2M

As for the smart phone application, a container was made to contain multiple application users. Discussed later, every user can define one’s own user-id on the smart phone app. The ”User” container of the AE that has been assigned to the SmartCampus application on OneM2M contains containers of every user. In the container of a user, two nested containers are defined, one for the GPS data of the user and one for the status of the user. To clarify, the structure for SmartCampus users can be seen in Figure 80.

HTTP connection (Smartphone to HUAWEI platform):

For the connection with HUAWEI IoT platform, the smartphone app publishes an http post whenever a new GPS location is generated (at 1 Hz).

The smartphone app sends the same data which is also posted in the “con” field to OneM2M:

"{"type":<type>,"id":<userID>, "timestampUtc":<GPSTimestampUnixUTC>, "lon":<userLongitude>, "lat":<userLatitude>, "speed":<userSpeed>, "heading":<userHeading>, “Accuracy”:<userAccuracy>}"

Currently this is posted to HUAWEI servers on the following address: http://217.110.131.79:2020/mobile/dataapp

In this http post, only a header is available: "Content-Type", "application/json".

The smartphone application stores the response from HUAWEI server on the smartphone in the same format as stored on the HUAWEI server.

8.3 NB-IoT device

The NB-IoT device gets data from a puddle detector and sends it to the TIM ICON oneM2M platform by means of the communication scheme shown in Figure 81.

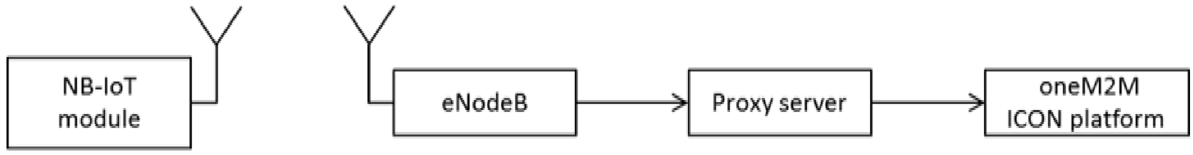

Figure 81 – Communication scheme between NB-IoT device and ICON oneM2M platform.

The NB-IoT module sends an UDP message where the destination IP is the one corresponding to the proxy server. The payload of the message contains the hexadecimal conversion of the ASCII characters composing the JSON format of the data. Once the proxy server receives the UDP message from the eNodeB, it converts the payload again in the ASCII format. Then the proxy server forwards the message to the TIM oneM2M ICON platform. Figure 82 shows how the information carried by the message is stored in the oneM2M platform.

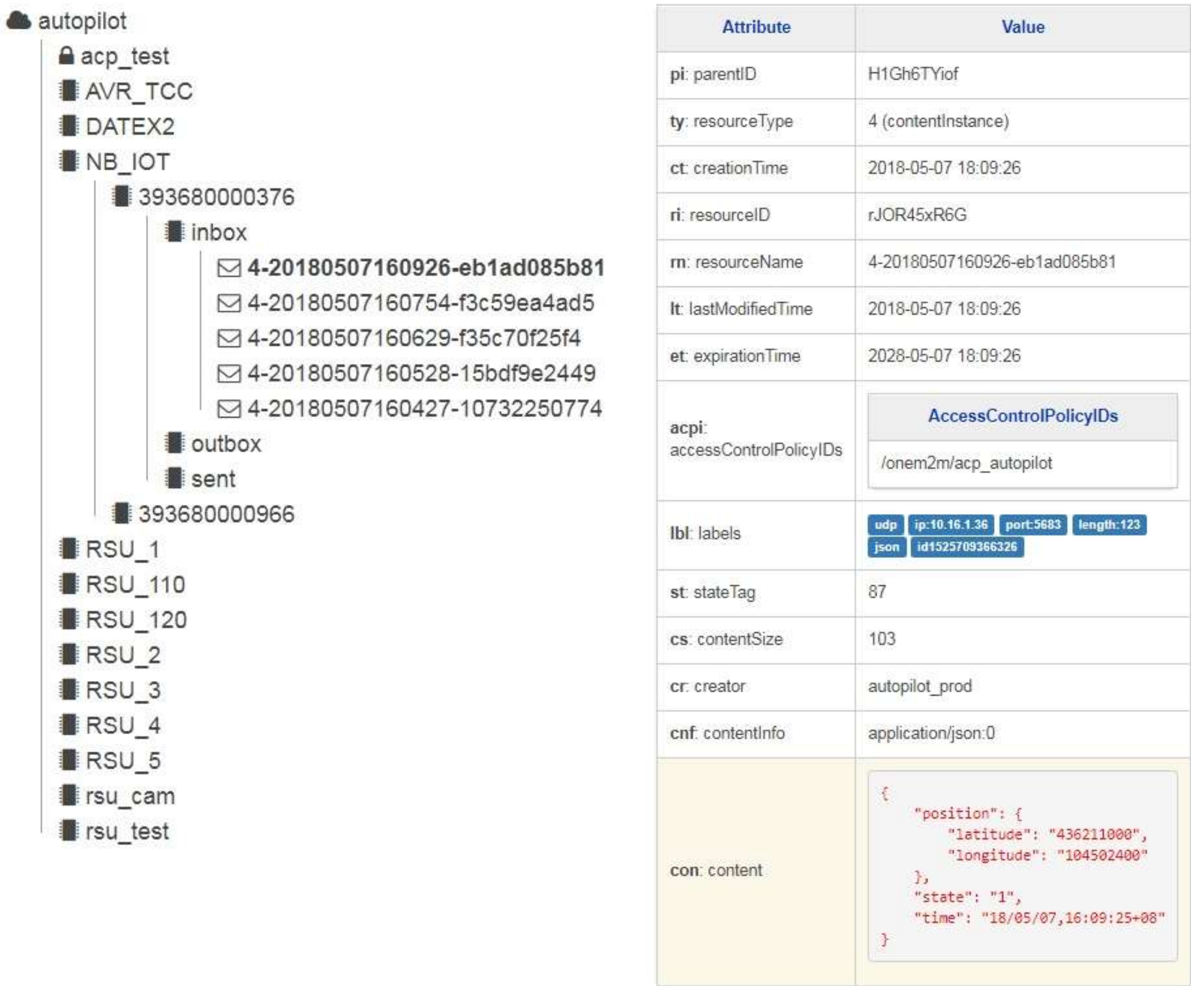

Figure 82 – TIM ICON oneM2M platform resource tree.

The "autopilot" application entity includes a "NB_IOT" container. It collects data coming from two SIM, respectively identified by "393680000376" and "393680000966" numbers. The one placed in the tested NB-IoT device is associated to the number 393680000376. The corresponding sub-container includes the "inbox" folder, where each content instance stores the information related to a single NB-IoT device transmission. For example, the table on the right in Figure 82 shows the information collected by the “4-20180507160926-eb1ad085b81” content instance. The value of the “con” field (the one highlighted in red) is the JSON format of the data carried by the payload of the corresponding UDP message sent by the NB-IoT device. As can be seen, it is constituted by three kinds of information: device position, puddle detector state and timestamp.

9. Conclusions and discussions

This document describes the development and integration of IoT devices used in different AUTOPILOT use cases to support autonomous driving functions. IoT sensor devices and mobile IoT objects (mobile robots and/or micro aerial vehicles) are integrated with the IoT infrastructure (sensor/actuators, connectivity and communication) developed for different use cases and seamlessly deployed into the five pilot sites that are forming various IoT ecosystems including other IoT devices, vehicle IoT platforms developed in T2.1 and Open IoT platform developed in T2.3.

Five different IoT platforms have been used for collecting, exchanging and processing the data from the IoT devices in the different use cases in the AUTOPILOT pilot sites:

- FIWARE IoT platform, used in the Dutch pilot site;
- Watson IoT Platform, used in the Dutch and Spanish pilot sites;
- HUAWEI OceanConnect IoT platform, used in the Dutch pilot site;
- TIM oneM2M IoT platform, used in the Italian pilot site;
- SENSINOV oneM2M platform, used in the Finnish, French and Dutch pilot sites.

The development and integration of IoT devices into IoT ecosystem are adapted to the pilot sites infrastructure. The use cases map the AUTOPILOT architecture and the IoT devices are integrated into different architecture components and interfaced/connected to the use case and pilot site IoT platform (infrastructure, connectivity, services, etc.). IoT devices used in different AUTOPILOT use cases support/enhance the autonomous driving functions. The IoT devices used are adapted to the autonomous driving function requirements in terms of speed of access (latency), availability and range (covered area).

The IoT devices used in the use cases have some localised mission-critical functions, such as warning other vehicles in the immediate proximity that a pedestrian is jaywalking, which need to be accessible within very low latency. IoT devices are supporting autonomous driving functions, such as notification about a parking spot being made available, which require that the IoT device connectivity covers wider areas and also require less latency.

The objective of using IoT devices is to support and enhance the autonomous driving functions with respect to SAE levels.

The vehicle used in the different AUTOPILOT use cases starts at level 2 with internal systems that take care of the different aspects of driving such as steering, acceleration and braking. The driver is able to intervene if any part of the vehicle system fails. Examples of level 2 include use cases helping vehicles to stay in lanes and self-parking features, with more than one ADAS aspect. Tesla's Autopilot, Nissan's ProPilot are example of level 2, as the vehicles can automatically keep you in the right lane on the road and keep you at a safe distance from the vehicle in front when in a traffic jam.

Level 3 vehicles are ones that can be truly considered autonomous. Drivers are allowed to safely use their phone or watch movies, although they are still required to be on-hand to intervene if necessary. The vehicles having implemented level 3 technology are able to detect the environment around them, and the vehicles can make own informed decisions such as overtaking slower moving vehicles. Audi's A8L is an example of vehicle with level 3 autonomy (the vehicle takes over all aspects of driving, in slow-moving traffic up to 60kph).

Level 4 vehicles are able to take decisions so the driver isn't required to intervene. The restrictions for level 4 are that the full self-driving mode can only be activated in certain, geofenced areas or in traffic jams.

The main difference between level 3 and level 4 automation is that level 4 vehicles are able to intervene themselves if things go wrong or there is a system failure. The vehicles using level 4 are dependent on their own devices without any human intervention in the most of situations, but there is an option to manually override in difficult or preferable circumstances. Google's Waymo is an example of vehicle with level 4 autonomy operating driver free, although a test driver is on hand just in case anything goes wrong.

According to the AUTOPILOT partners, the IoT devices and technologies can support the autonomous driving functions as following:

- Dutch PS (Urban driving/Car rebalancing): In best case, the aim is to increase from AD level 3 to a maximum of level 4;
- Italian PS (Highway pilot): IoT technology assists the rising of the automation level from 3 to 4;
- Italian PS (Urban driving): The IoT technology enhances the rising of the automation level from 3 to 4;
- Spanish PS (Automated valet parking): Allowing the AD function to increase the automation level from 3 to 4;
- Spanish PS (Urban driving): IoT technology allows increasing the automation level from 3 to 4;
- Finnish PS (Urban driving): The inclusion and use of IoT technology will move the automation level from 3 to 4, limited to the operational design domain (ODD) automatic valet parking allowing driverless parking of the vehicle.

The estimations at this stage are optimistic. The development and integration of IoT devices into combined autonomous vehicles and IoT ecosystems could support the integration of services using interoperable IoT platforms and IoT devices that provide additional information to the vehicles about the environment, surroundings and the dynamic events around the vehicles.

10. References

[1] A. Schmeitz, S. Jansen (editors). *Initial specification of IoT-enabled autonomous driving use cases*. AUTOPILOT project, June 2017.

[2] G. Solmaz, M. Almeida (editors). *Initial IoT self-organizing platform for self-driving vehicles*. AUTOPILOT project, June 2017.

[3] V. Filippo, L. Altomare (editors). *Initial open IoT vehicle platform specification*. AUTOPILOT project, October 2017.

[4] Telecom Italia SPA (editor). *Initial specification of communication system for IoT enhanced AD*. AUTOPILOT project, September 2017.

[5] M.S. Greco. *Automotive Rada*r. IEEE Radar Conference, Atlanta, May 2012.

[6] R.E. Makhour. *Renesas Autonomy - The open platform for automated driving*. Electronics World, June 2017.

[7] Frost & Sullivan. *LiDAR: Driving the Future of Autonomous Navigation - Analysis of LiDAR technology for advanced safety*, 2016. Online at: http://velodynelidar.com/docs/papers/FROST-ON-LiDAR.pdf

[8] O. Vermesan and P. Friess (Eds.). Digitising the Industry - Internet of Things Connecting the Physical, Digital and Virtual Worlds, ISBN: 978-87-93379-81-7, River Publishers, Gistrup, 2016.

[9] O. Vermesan and J. Bacquet (Eds.). Cognitive Hyperconnected Digital Transformation Internet of Things Intelligence Evolution, ISBN: 978-87-93609-10-5, River Publishers, Gistrup, 2017.

[10] O. Vermesan, R. Bahr, A. Gluhak, F. Boesenberg, A. Hoeer and M. Osella, "IoT Business Models Framework", 2016, online at http://www.internet-of-things-research.eu/pdf/D02_01_WP02_H2020_UNIFY-IoT_Final.pdf.

[11] M. Almeida and al., "Initial Specification of Communication System for IoT Enhanced Autonomus Driving", AUTOPILOT, 2017 online at: http://autopilot-project.eu/.

[12] The European Research Cluster on the Internet of Things (IERC) (http://www.internet-ofthings-research.eu).

[13] X. Wu et al., "Cars Talk to Phones: A DSRC Based Vehicle-Pedestrian Safety System," 2014 IEEE 80th Vehicular Technology Conference (VTC2014-Fall), Vancouver, BC, 2014, pp. 1-7.

[14] Istituto Superiore Mario Boella, LIMPID project, smart mobility and Smart Data Net https://vimeo.com/156536073

[15] The Ko-TAG project - https://www.iis.fraunhofer.de/en/ff/lv/lok/proj/kotag.html

[16] Bosch Motorcycle-to-vehicle communication - https://www.youtube.com/watch?v=BXXlodI9gO0

[17] M. Bagheri, M. Siekkinen and J. K. Nurminen, "Cellular-based vehicle to pedestrian (V2P) adaptive communication for collision avoidance," 2014 International Conference on Connected Vehicles and Expo (ICCVE), Vienna, 2014, pp. 450-456.

[18] J. J. Anaya, P. Merdrignac, O. Shagdar, F. Nashashibi and J. E. Naranjo, "Vehicle to pedestrian communications for protection of vulnerable road users," 2014 IEEE Intelligent Vehicles Symposium Proceedings, Dearborn, MI, 2014, pp. 1037-1042.

[19] A. Lewandowski, S. Bocker, V. Koster and C. Wietfeld, "Design and performance analysis of an IEEE 802.15.4 V2P pedestrian protection system," 2013 IEEE 5th International Symposium on Wireless Vehicular Communications (WiVeC), Dresden, 2013, pp. 1-6.

[20] M. Carignani, S. Ferrini, M. Petracca, M. Falcitelli and P. Pagano, "A prototype bridge between automotive and the IoT," *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, Milan, 2015, pp. 12-17. doi: 10.1109/WF-IoT.2015.7389019.

[21] “ETSI EN 302 636-5-1 V2.1.0 (2017-05) Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 5: Transport Protocols; Sub-part 1: Basic Transport Protocol”

[22] "FI-WARE NGSI-9 Open RESTful API Specification", FIWARE Forge, 2017, to be retrieved via, https://forge.fiware.org/plugins/mediawiki/wiki/fiware/index.php/FI-WARE_NGSI-9_Open_RESTful_API_Specification.

[23] "FI-WARE NGSI-10 Open RESTful API Specification", FIWARE Forge, 2017, to be retrieved via, https://forge.fiware.org/plugins/mediawiki/wiki/fiware/index.php/FI-WARE_NGSI-10_Open_RESTful_API_Specification.

[24] Carlino A., Altomare L., Darin M., Visintainer F., Marchetto A. (2016) Automotive LIDAR-Based Strategies for Obstacle Detection Application in Rural and Secondary Roads. In: Schulze T., Müller B., Meyer G. (eds) Advanced Microsystems for Automotive Applications 2015. Lecture Notes in Mobility. Springer, Cham

[25] *Automated Driving Systems 2.0 - A Vision for Safety*. U.S. Department of Transportation and National Highway Traffic Safety Administration. DOT HS 812 442, September 2017. Online at: https://www.nhtsa.gov/sites/nhtsa.dot.gov/files/documents/13069a-ads2.0_090617_v9a_tag.pdf

[26] Création d'applications mobiles - Beacon et Eddystone. Blue Net Beacon. Online at (French): http://www.bnbeacon.com/

[27] N. Dalal and B. Triggs , Histograms of oriented gradients for human detection, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005.