Grant Agreement Number: 731993

Project acronym: AUTOPILOT

Project full title: AUTOmated driving Progressed by Internet Of Things

D6.10

FINAL VERSION OF THE DATA MANAGEMENT PLAN

Due delivery date: 31/12/2019

Actual delivery date: 26/12/2019

Organization name of lead participant for this deliverable: AKKA

Dissemination level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential , only for members of the consortium (including the Commission Services)	

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731993.

Document Control Sheet

Deliverable number:	D6.10
Deliverable responsible:	AKKA
Workpackage:	6
Editor:	AIT AAZIZI Amine – ZOUGARI Sadeq

Author(s) – in alphabetical order		
Name	**Organisation**	**E-mail**
DALET Benoit	AKKA	benoit.dalet@akka.eu
ZOUGARI Sadeq	AKKA	Sadeq.zougari@akka.eu

Document Revision History			
Version	**Date**	**Modifications Introduced**	
		Modification Reason	**Modified by**
V0.1	20/07/2019	First Draft	DALET, Benoit ; ZOUGARI, Sadeq
V0.2	02/12/2019	Peer review	KOVACS, Ernoe ; SIMEON, Jean-François
V1.0	31/12/2019	Final version	DALET, Benoit ; ZOUGARI, Sadeq

Abstract

This document presents the final version of the data management plan, which identifies and describes all data collected and generated during the AUTOPILOT project and defines the data management principles followed by the Consortium.

Abbreviations and Acronyms

Acronym	Definition
ADAS	Advanced Driver-Assistance Systems
AVP	Automated Valet Parking
CAN	Controller Area Network
DMP	Data Management Plan
EC	European Commission
EC	European Commission
FAIR	Findable – Accessible – Interoperable-Reusable
FESTA	Field opErational teSt supporT Action
GA	Grant Agreement
GPS	Global Positioning System

IOT	Internet Of Things
ITS	Intelligent Transportation Systems
LIDAR	LIght Detection And Ranging
ORDP	Open Research Data Pilot
PO	Project Officer
RADAR	RAdio Detection And Ranging
RPM	Revolutions Per Minute
WP	Work Package
WPL	Work Package Leader

Table of contents

List of Figures

List of Tables

Executive Summary

In Horizon 2020 a limited pilot action on open access to research data has been implemented. Participating projects have been required to develop a Data Management Plan (DMP).

This deliverable provides the third version of the Data Management Plan elaborated by the AUTOPILOT project. The purpose of this document is to provide an overview of the main elements of the data management policy. It outlines how research data was handled during the AUTOPILOT project and describes what data was collected, processed or generated and following what methodology and standards, whether and how this data was shared and/or made open, and how was be curated and preserved. Besides, data types list, metadata and global data collection processes are also defined in this document.

The AUTOPILOT Data Management Plan refers to the latest EC DMP guidelines[1]. This version has explicit recommendations for full life cycle management through the implementation of the FAIR principles, which state that the data produced shall be Findable, Accessible, Interoperable and Reusable (FAIR).

The document first provide a kind of ontology for the data used during the AUTOPILOT projects, generated by a diversity of data sources, like vehicles, on board and road side sensors. The report presents the hierarchical IoT architecture, the different data set categories and the metadata, which are important to interpret the data content in the IoT platforms.

The report includes a detailed presentation of the methodology for the AUTOPILOT data management, taking into account the decision of the AUTOPILOT consortium to apply for the ORDP.

The report provides the description of the data sets structures, relating to the different categories of IoT devices and other data sources like vehicles, but also the specific types and data set used at the different pilot sites, according to the use cases deployed and assessed locally.

[1] http://ec.europa.eu/research/participants/data/ref/h2020/grants_manual/hi/oa_pilot/h2020-hi-oa-data-mgt_en.pdf

1. Introduction

1.1 Objectives of the project

Automated driving is expected to increase safety, to provide more comfort and to create many new business opportunities for mobility services. The Internet of Things (IoT) is about enabling connections between objects or "things"; it is about connecting anything, anytime, anyplace, using any service over any network.

AUTOmated Driving **P**rogressed by **I**nternet **O**f **T**hings" (AUTOPILOT) project will especially focus on utilizing the IoT potential for automated driving.

The overall objective of AUTOPILOT is to bring together relevant knowledge and technology from the automotive and the IoT value chains in order to develop IoT-architectures and platforms which will bring Automated Driving towards a new dimension. This is realized through the following main objectives:

- Use, adapt and innovate current and advanced technologies to define and implement an IoT approach for autonomous and connected vehicles
- Deploy, test and demonstrate IoT based automated driving use cases at several permanent pilot sites, in real traffic situations with: Urban driving, Highway pilot, Automated Valet Parking, Platooning and Real-time car sharing.
- Create and deploy new business products and services for fully automated driving vehicles, used at the pilot sites: by combining stakeholders' skills and solutions, from the supply and demand side
- Evaluate with the involvement of users, public services and business players at the pilot sites :
  - The suitability of the AUTOPILOT business products and services as well as the ability to create new business opportunities
  - The user acceptance related to using the Internet of Things for highly or fully automated driving
  - The impact on the citizens' quality of life
- Contribute actively to standardization activities as well as to consensus building in the areas of Internet of Things and communication technologies

Automated vehicles largely rely on on-board sensors (LiDAR, radar, cameras, etc.) to detect the environment and make reliable decisions. However, the possibility of interconnecting surrounding sensors (cameras, traffic light radars, road sensors, etc.) exchanging reliably redundant data may lead to new ways to design automated vehicle systems potentially reducing cost and adding detection robustness.

Indeed, many types of connected objects may act as an additional source of data, which will very likely contribute to improve the efficiency of the automated driving functions, enable new automated driving scenarios as well as increase the automated driving function safety while providing driving data redundancy and reducing implementation costs. These benefits will enable pushing the SAE level of driving automation to the full automation, keeping the driver out of the loop. Furthermore, by making autonomous cars a full entity in the IoT, the AUTOPILOT project enables developers to create IoT/AD services as easy as accessing any entity in the IoT.

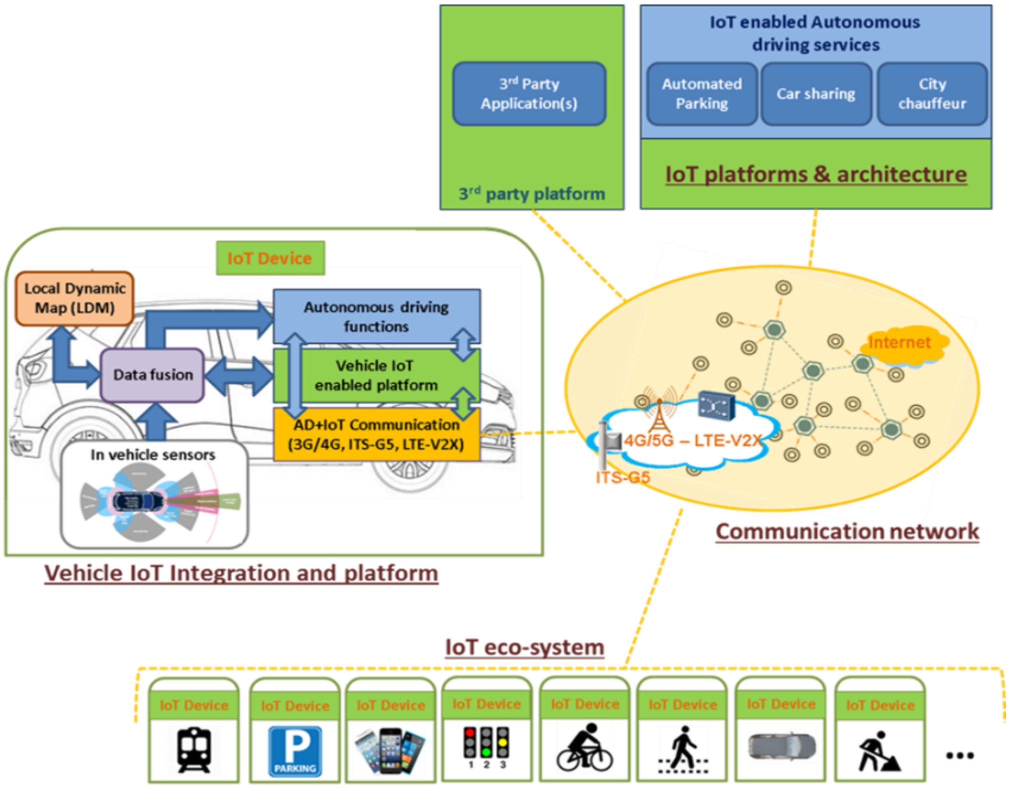

Figure 1 – The AUTOPILOT overall concept

The Figure above depicts the AUTOPILOT overall concept including the different ingredients to apply IoT to autonomous driving:

- The overall IoT platforms and architecture, allowing the use of the IoT capabilities for autonomous driving.
- The Vehicle IoT integration and platform to make the vehicle an IoT device, using and contributing to the IoT.
- The Automated Driving relevant sources of information (pedestrians, traffic lights …) becoming IoT devices and extending the IoT eco-systems to allow enhanced perception of the driving environment on the vehicle.
- The communication network using appropriate and advanced connectivity technology for the vehicle as well as for the other IoT devices.

1.2 Purpose of the document

This deliverable presents the third version of the data management plan elaborated for the AUTOPILOT project. The purpose of this document is to provide an overview of the data set types present in the project and to define the main data management policy adopted by the Consortium.

The data management plan defines how data in general and research data in particular are handled during the research project.

It describes what data is collected, processed or generated by the IOT devices and by all the IoT ecosystem, what methodologies and standards shall be followed during the collection process, whether and how this data is shared and/or made open not only for the evaluation needs but also to comply with the ORDP requirements[2], and how it shall be curated and preserved. Besides, the Data management plan identifies the four (4) key processes requirements that define the data collection process and provides first recommendations to be applied.

The document is structured as the following. **The Chapter 2** outlines a data overview in the AUTOPILOT project. It details AUTOPILOT data categories, data types and metadata, then the data collection processes to be followed and finally the test data flow and test data architecture environment.
The Chapter 3 gives a global vision of the test data management methodology developed in the WP3 across pilot sites.
The Chapter 4 gives insights about the Open Research Data Pilot under H2020 guidelines.
The Chapter 5 provides detailed description of the datasets used in AUTOPILOT project with focus on used methodologies, standard and data sharing policies.
The Chapter 6 gives insights about the FAIR Data Management principle under H2020 guidelines and how AUTOPILOT started actions in order to be FAIR compliant.
Finally, the remaining chapters outline the necessary roles, responsibilities and ethical issues

1.3 Intended audience

The AUTOPILOT project addresses highly innovative concepts. As such, foreseen intended audience of the project is the scientific community interested in the IOT and/or automotive technologies. In addition, due to the strong expected impact of the project on their respective domains, the other expected audience consists in automotive industrial communities, telecom operator and standardization organizations.

[2] https://www.openaire.eu/ordp/

2. Data in AUTOPILOT: an overview

The aim of this chapter is:

- To provide a first categorization of the data.
- To identify a list of the data types that is generated.
- To provide a list of metadata that is used to describe generated data and enable data re-use.
- To provide recommendations on data collection and sharing processes during the project and beyond.

2.1 The AUTOPILOT hierarchical IoT architecture

The AUTOPILOT project will collect a large amount of raw data to measure the benefit of IoT for automated driving with multiple automated driving use cases and services, at different pilot locations.

Data from vehicles and sensors is collected and managed through a hierarchy of IoT platforms as illustrated in Figure 2

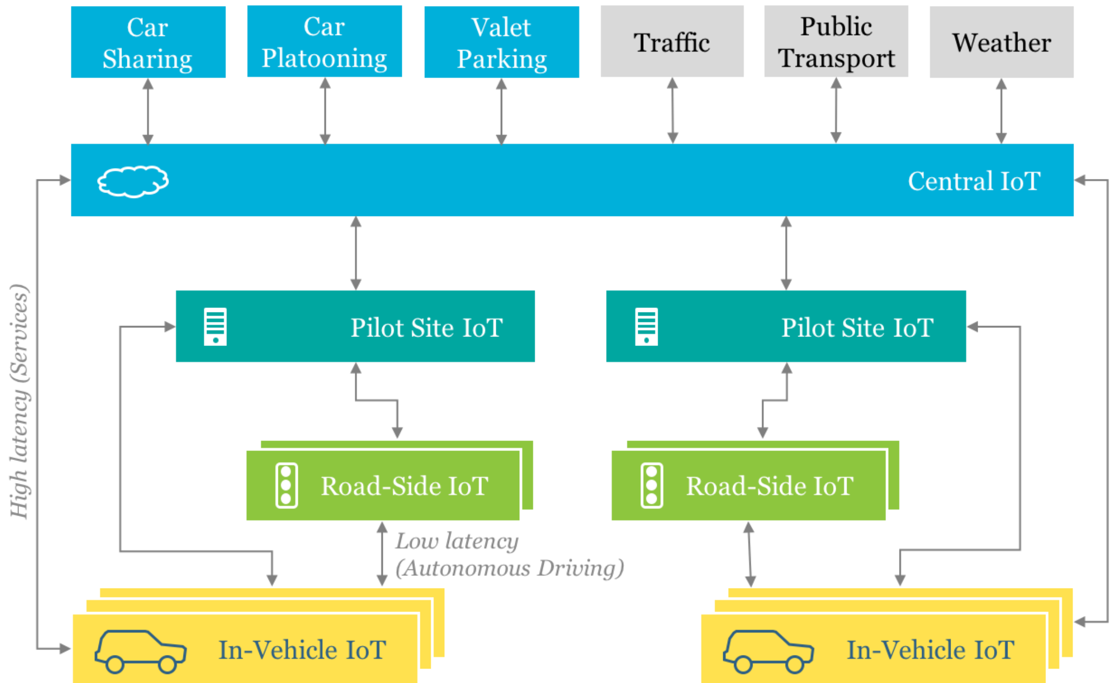

Figure 2 – The AUTOPILOT hierarchical IoT architecture

The figure above shows a federated architecture with the following four layers:

- **In-vehicle IoT Platforms:** Here is everything that is mounted inside the vehicle, i.e., components responsible for AD, positioning, navigation, real time sensor data analysis, and communication with the outside world. All mission critical autonomous driving functions should typically reside in this layer.
- **Road-Side IoT Platforms:** Road-side and infrastructure devices, such as cameras, traffic light sensors, etc., are integrated and managed as part of road-side IoT PFs

covering different road segments and using local low latency communication networks and protocols as required by the devices and their usage.

- **Pilot Site IoT Platforms:** This layer constitutes the first integration level. It is responsible for collecting, processing and managing data at the pilot sites level.
- **Central IoT Platform:** This is a Cloud-based top layer that integrates and aggregates data from the various pilot sites as well as external services (weather, transport, etc.). This is where the common AD services such as car sharing, platooning, etc., will reside. Data, at this level, are standardized using common formats, structures and semantics. The central IoT platform is hosted on IBM infrastructure.

The data analysis is performed according to Field Operational Test studies (FOT[3]) and using FESTA[4] methodology. The FESTA project funded by the European Commission developed a handbook on FOT methodology which gives general guidance on organizational issues, methodology and procedures, data acquisition and storage, and evaluation.
From raw data a large amount of derived data is produced to address multiple research needs. Derived data will follow a set of transformations: cleaning, verification, conversion, aggregation, summarization or reduction.
In any case, data must be well documented and referenced using rich metadata in order to facilitate and foster sharing, to enable validity assessments and to enable its usage in an efficient way.
Thus, each data must be described using additional information called metadata. The latter must provide information about the data source, the data transformations and the conditions in which the data has been produced. More details about the metadata in AUTOPILOT are described in section 2.2.

2.2 Data sets categories

The AUTOPILOT project will produce different categories of data sets.

- **Context data**: data that describe the context of an experiment (e.g. Metadata)
- **Acquired and derived data**: data that contain all the collected information from measurements and sensors related to an experiment.
- **Aggregated data**: data summary obtained by reduction of acquired data and generally used for data analysis.

2.2.1 Context data

Context data is any information that helps to explain observation during a study. Context data can be collected, generated or retrieved from existing data. For example, it contains information such as vehicle, road or drivers characteristics.

2.2.2 Acquired and derived data

Acquired data is all data collected to be analysed during the course of the study. Derived data is created by different types of transformations including data fusion, filtering, classification, and reduction. Derived data are easy to use and they contain derived measures and performance indicators referring to a time period when specific conditions are met. This category includes measures from sensors coming from vehicles or IOT and

[3] http://fot-net.eu/
[4] http://wiki.fot-net.eu/index.php/FESTA_Handbook

subjective data collected from either the users or the environment.

The following list outlines the data types and sources that is collected:

In-vehicle measures are the collected data from vehicles, either using their original in-car sensors or sensors added for AUTOPILOT purposes. These measures can be divided into different types:	
	Vehicle dynamics are measurements that describe the mobility of the vehicle. Measurements can be for example longitudinal speed, longitudinal and lateral acceleration, yaw rate, and slip angle.
	Driver actions define the driver actions on the vehicle commands that can be measured; for instance, steering wheel angle, pedal activation or HMI button press variables, face monitoring indicators characterizing the state of the driver, either physical or emotional.
	In-vehicle systems state can be accessed by connecting to the embedded controllers. It includes continuous measures like engine RPM or categorical values like ADAS and active safety systems activation.
	Environment detection is the environment data that can be obtained by advanced sensors like RADARs, LIDARs, cameras and computer vision, or more simple optical sensors. For instance, luminosity or presence of rain, but also characteristics and dynamics of the infrastructure (lane width, road curvature) and surrounding objects (type, relative distances and speeds) can be measured from within a vehicle.
	Vehicle positioning the geographical location of a vehicle is determined with satellite navigation systems (e.g. GPS) and the aforementioned advanced sensors.
	Media mostly consist of video. The data consist of media data but also index files used to synchronize the other data categories. They are also often collected from the road side.
Continuous subjective measures Complimentary to sensors and instrumentation, some continuous measures can also be built in a more subjective way, by analysts or annotators, notably using video data.	
Road-side measures are the vehicle’s counting speed measurement and positioning, using radar, rangefinders, inductive loops or pressure hose. In ITS systems, it may also contain more complex information remotely transferred from vehicles to road-side units.	
Experimental conditions are the external factors which may have an impact on participants’ behaviour. They may be directly collected during the experiment, or integrated from external sources. Typical examples are traffic density and weather conditions.	
IOT data are the external sources of data that is collected/shared through IOT services:	
	Users Data can be generated by smartphones or wearables. The users can be the pedestrians or the car drivers. These data helps the user experience for the usage of services by vehicle or infrastructure. The privacy aspects are well explained in chapter 4.
	Infrastructure Data are all the data giving additional information about the environment. Typical examples are the traffic status, road-works, accidents and road conditions. They can also be directly collected from Road-side cameras or traffic light control units and then transferred to IOT Platforms. For instance, the infrastructure data can transfer hazard warnings or expected occupancy of busses on bus lanes to vehicles using communication networks.

	In-Vehicle data defines the connected devices or sensors in vehicles. Typical examples are navigation status, time distance computations, real-time pick-up / drop-off information for customers, and events detected by car to be communicated to other vehicles or GPS data to be transferred to maps.
Surveys data are data resulting from the answers of surveys and questionnaires for user acceptance evaluation	

2.2.3 Aggregated data

Aggregated data is generally created in order to answer the initial research question. They are supposed to be verified and cleaned, thus facilitating their usage for analysis purposes.
Aggregated data contains a specific part of the acquired or derived data (e.g. the average speed during a trip or the number of passes through a specific intersection). Its smaller size allows a simple storage in e.g. database tables and an easy usage suitable for data analysis. To obtain aggregated data, several data reduction processes are performed. The reduction process summarizes the most important aspects in the data into a list of relevant parameters or events, through one or all of the following processes: validation, curation, conversion, annotation.
Besides helping in answering new research questions, aggregated data may be re-used with different statistical algorithms without the need to use raw data. For AUTOPILOT, aggregated data will represent the most important data types that is shared by the project as described in section 4 Participation in the open research data pilot. It does not allow potentially problematic re-uses because it does not contain instantaneous values that would highlight illegal behaviour of a vehicle, a driver or another subsystem.

2.3 Metadata

2.3.1 General principles

This section reviews the relevant metadata standards developed or used in the previous and ongoing field operational tests (FOT) and naturalistic driving studies (NDS) as a basis for the development of the metadata specifications of the pilot data. Such standards will help the analysis and re-use of the collected data within the AUTOPILOT project and beyond.

The text in this section is derived from the work done in the FOT-Net Data project [5] for sharing data from field operational tests. The results of this work are described in the Data Sharing Framework[6]. The CARTRE project [7] is currently updating this document to specifically addressing road automation pilots and FOTs.

As described in the previous sections, the pilots will generate and collect a large amount of raw and processed data from continuous data-logging, event-based data collection, and surveys. The collected data is analysed and used for various purposes in the project including the impact assessment carried out by partners who are not involved in the pilots. This is a typical issue encountered in many FOT/NDS projects in which the data analyst (or re-user) needs to know how the raw data was collected and processed in order to perform data

[5] http://fot-net.eu
[6] http://fot-net.eu/Documents/d3-1-data-sharing-framework/
[7] http://connectedautomateddriving.eu

analysis, modelling and interpretation. Therefore, good metadata is vital.

The Data Sharing Framework defines metadata as '**any information that is necessary in order to use or properly interpret data**'. The aim of this section is to address these issues and to provide methods to efficiently describe a dataset and its associated metadata. It results in suggestions for good practices in documenting a data collection and datasets in a structured way. Following the definition of metadata by the data sharing framework, we divide the AUTOPILOT's Metadata into four different categories as follows.

- **AUTOPILOT pilot design and execution** documentation, which corresponds to a high level description of a data collection: its initial objectives and how they were met, description of the test site, etc.
- **Descriptive** metadata, which describes precisely each component of the dataset, including information about its origin and quality.
- **Structural** metadata, which describes how the data is being organized.
- **Administrative** metadata, which sets the conditions for how the data can be accessed and how this is being implemented.

Field Operational Tests (FOTs) have been carried out worldwide and adopted different metadata formats to manage the collected data. One of good examples is the ITS Public Data Hub hosted by the US Department of Transport[8]. There are over 100 data sets created using ITS technologies. The data sets contain various types of information, such as highway detector data, travel times, traffic signal timing data, incident data, weather data, and connected vehicle data, many of them will also be collected in the AUTOPILOT data. The ITS Public Data Hub uses ASTM 2468-05 standard format for metadata to support archived data management systems. This standard would be a good start point to design metadata formats for various types of operational data collected by the IoT devices and connected vehicles in AUTOPILOT.

In a broader context of metadata standardisation, there are a large number of metadata standards available which address the needs of particular user communities. The Digital Curation Centre (DCC) provides a comprehensive list of metadata standards [9] for various disciplines such as general research data, physical science as well as social science & humanities. It also lists software tools that have been developed to capture or store metadata conforming to a specific standard.

2.3.2 IOT metadata

The metadata describing IoT data are specified in the context of OneM2M standard[10]. In such context "Data" signifies digital representations of anything. In practice, that digital representation is associated to a "container" resource having specific attributes. Those attributes are both metadata describing the digital object itself, and the values of the variables of that object, which are called "content".

[8] https://catalog.data.gov/dataset
[9] http://www.dcc.ac.uk/resources/metadata-standards/list
[10] http://www.onem2m.org/

Every time an IoT device publishes new data on the OneM2M platform a new "content instance" is generated, representing the actual status of that device. All the "content instances" are stored in the internal database with a unique resource ID.

The IOT metadata are describing the structure of the information, according to the OneM2M standard. The IoT metadata are described in the Table 1below.

Table 1 – OneM2M Metadata for IoT data [11]

Metadata Element	Extended name	Description
pi	parentID	ResourceID of the parent of this resource.
ty	resourceType	Resource Type attribute identifies the type of the resource as specified in clause. For E.g. "4 (contentInstance)".
ct	creationTime	Time/date of creation of the resource. This attribute is mandatory for all resources and the value is assigned by the system at the time when the resource is locally created. Such an attribute cannot be changed.
ri	resourceID	This attribute is an identifier for the resource that is used for 'non-hierarchical addressing method', i.e. this attribute contains the 'Unstructured-CSE-relative-Resource-ID' format of a resource ID as defined in table 7.2-1 of [5]. This attribute is provided by the Hosting CSE when it accepts a resource creation procedure. The Hosting CSE assign a resourceID which is unique in that CSE.
rn	resourceName	This attribute is the name for the resource that is used for 'hierarchical addressing method' to represent the parent-child relationships of resources. See clause 7.2 in [5] for more details.
lt	lastModifiedTime	Last modification time/date of the resource. The lastModifiedTime value is updated when the resource is updated.
et	expirationTime	Time/date after which the resource is deleted by the Hosting CSE.
acpi	accessControlPolicyIDs	The attribute contains a list of identifiers of an <accessControlPolicy> resource. The privileges defined in the <accessControlPolicy> resource that are referenced determine who is allowed to access the resource containing this attribute for a specific purpose (e.g. Retrieve, Update, Delete, etc.).
lbl	label	Tokens used to add meta-information to resources. This attribute is optional. The value of the labels attribute is a list of individual labels, that can be used for example for discovery

[11] AUTOPILOT D3.6 - Data collection and integration methodology

		purposes when looking for particular resources that one can "tag" using that label-key.
st	stateTag	An incremental counter of modification on the resource. When a resource is created, this counter is set to 0, and it is incremented on every modification of the resource
cs	contentSize	Size in bytes of the content attribute.
cr	creator	The ID of the entity (Application Entity or Common Services Entity) which created the resource containing this attribute
cnf	contentinfo	Information on the content that is needed to understand the content. This attribute is a composite attribute. It is composed first of an Internet Media Type (as defined in the IETF RFC 6838) describing the type of the data, and second of an encoding information that specifies how to first decode the received content. Both elements of information are separated by a separator defined in OneM2M TS-0004 [3].
or	ontologyRef	This attribute is optional. A reference (URI) of the ontology used to represent the information that is stored in the contentInstances resources of the <container> resource. If this attribute is not present, the contentInstance resource inherits the ontologyRef from the parent <container> resource if present.

3. Data management methodology in AUTOPILOT

The AUTOPILOT data collection process and data management is built upon requirements coming from 4 processes:

- **The evaluation requirement** defines the minimum data that must be collected in order to perform the evaluation process at the end of the project
- **The test specification** provides details about the data to be collected on the basis of the evaluation requirements and according to use cases specifications
- **The test data management** defines the data collection, harmonization, storage and sharing requirements using the first two processes and the ORDP process
- **The Open Research Data Pilot** [12] **(ORDP)** defines the requirement related to data sharing of research data

3.1 Evaluation process requirements

AUTOPILOT applies an evaluation methodology based on FESTA (Field opErational teSt supporT Action). This methodology consist of collecting relevant set of data to carry out the assessment of hypothesis relating to the benefit or improvement provided by IT solutions for the driving of vehicles, in the context of AUTOPILOT, Automated Driving.

The Figure 3 below shows a high level view of the data that is collected and integrated in the evaluation process. Different types of data (in blue) are collected, stored and analysed by different processes.

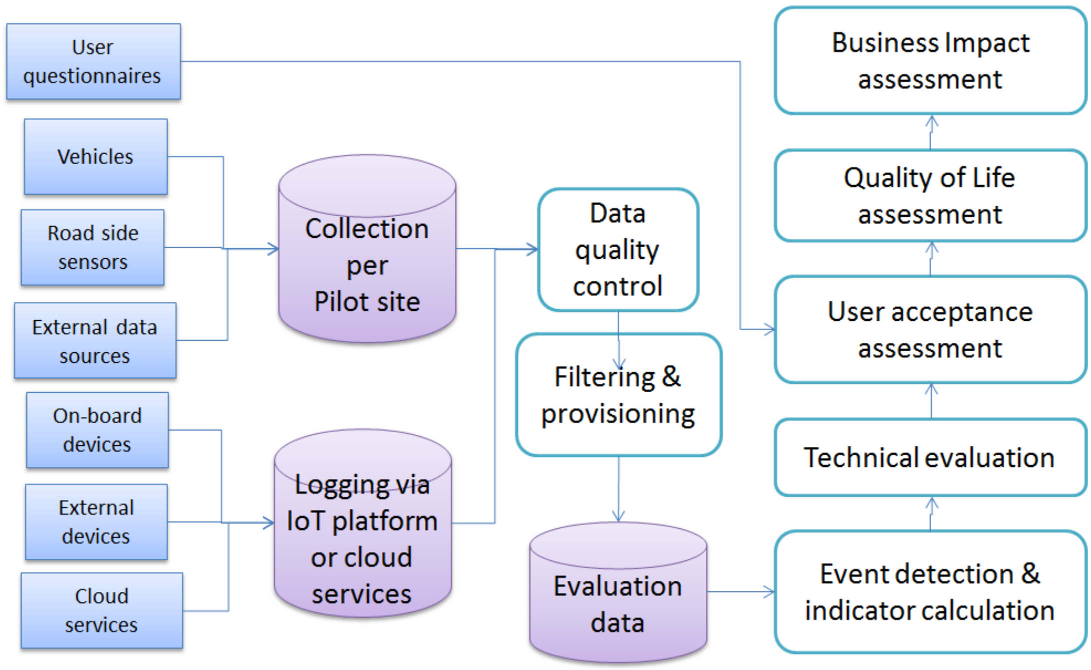

Figure 3 – Evaluation methodology

To fulfil the project objectives, a design of experiment is performed during the evaluation

[12] https://www.openaire.eu/opendatapilot

task. This design creates requirements that define the number of scenarios and test cases, the duration of tests and test runs, the number of situations per specific event, the number of test vehicles, the variation in users, the variation in situations (weather, traffic, etc.). Each pilot site must comply with this design of experiment and provide sufficient and meaningful data with the required quality level to enable technical evaluation.

3.2 Tests specification process requirements

The pilot tests specification Task T3.1 plays a major role that must be thoroughly followed. Indeed, this task will convert the high level requirements defined in the evaluation process into specific and detailed specifications of data formats, data size, data currencies, data units, data files, and storage. The list of requirements is defined for each of the following items: Pilot sites, Scenarios, Test Cases, Measures, Parameters, Data quality, etc. and is described in deliverable D3.1. All the development tasks of WP2 must implement completely, if impacted, the requirement defined in D3.1 in order to provide all the data (test data) as expected by the technical evaluation.

3.3 Open research data pilot requirement process

Additional requirements related to ORDP (Open Research Data Pilot) are defined in this document to guarantee that the collected data is provided in compliance to European Commission Guidelines[13] on Data Management in Horizon 2020. Those requirements are deeply defined and explained in chapter 4 .

3.4 Test data management methodology

The main objective of the data management plan is to define the methodology to be applied in AUTOPILOT across all pilot sites, in particular test data management. This includes the explanation of the common data collection and integration methodology.

One of the main objectives within T3.4 “Test Data management” is to ensure the comparability and consistency of collected data across pilot sites. In this context, the methodology is highly impacted by the pilot sites specifications of the Task 3.1 and compliant with the evaluation methodologies developed in the task 4.1. In particular, Technical evaluation primarily needs log data from the vehicles, IoT platforms, cloud services and situational data from pilot sites to detect situations and events, and to calculate indicators.

The log data parameters that are needed for technical evaluation are organized by data sources (vehicle sources, vehicle data, derived data, positioning, V2X messages, IoT messages, events, situations, surveys and questionnaires)

For IoT data, some pilot sites use proprietary IoT platforms in order to collect specific produced IoT data with specific devices or vehicles (e.g. the Brainport car sharing service and automated valet parking service use Watson IoT Platform™ to collect data from their vehicles).

On the top of that, we have OneM2M interoperability platform in each pilot site. This is the interoperability IoT platform for exchanging IoT messages relevant to all autonomous driving (AD) vehicles at pilot site level. Then, the test data is stored in pilot site test server storage

[13] Guidelines on FAIR Data Management in Horizon 2020 - Version 3.0 - 26 July 2016

that will contain mainly the vehicle data, IoT data and surveys data. Further, the test data is packaged and sent to the AUTOPILOT central storage that will allow evaluators access all the pilot site data in a common format. This includes the input from all pilot sites and use cases and for all test scenarios and test runs.

Every pilot site has its own test storage server for data collection (distributed data management), named PSTS (Pilot Site Test Server). In addition, there is a central storage server where data from all pilot sites is stored for evaluation and analysis, named CTS (Centralized Test Server).

Please note that the CTS and the PSTS are resources of the project, not available for public use.

The following figure represents the data management methodology and architecture used in AUTOPILOT across all pilot sites.

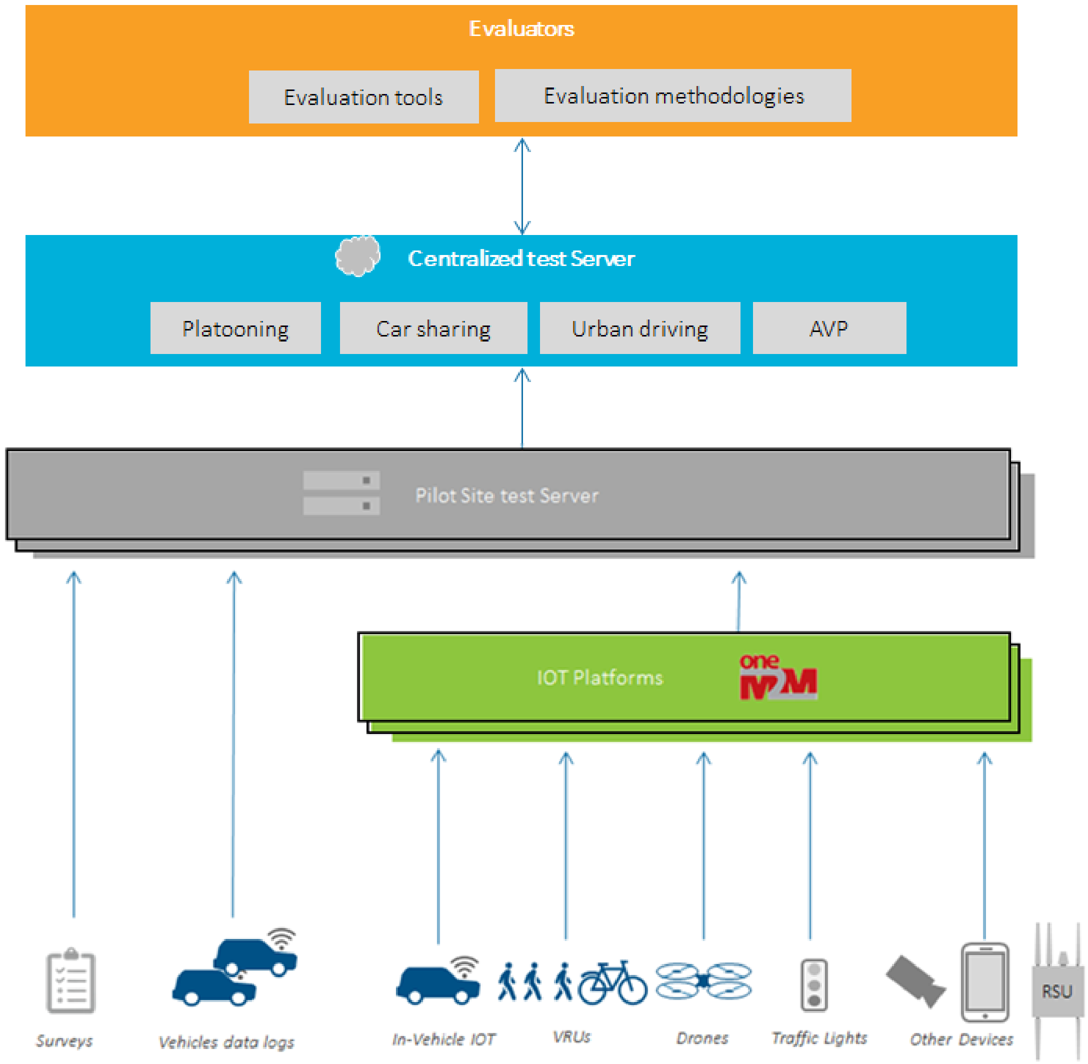

Figure 4 – Generic scheme of data architecture in AUTOPILOT

4. Participation in the open research data pilot

The AUTOPILOT project has agreed to participate in the Pilot on Open Research Data in Horizon 2020[14]. The project uses specific Horizon 2020 guidelines associated with 'open' access to ensure that the results of the project results provide the greatest impact possible.

AUTOPILOT will ensure the open access[15] to all peer-reviewed scientific publications relating to its results and will provide access to the research data needed to validate the results presented in deposited scientific publications.

The following lists the minimum fields of metadata that should come with an AUTOPILOT project-generated scientific publication in a repository:

- The terms: "European Union (EU)", "Horizon 2020"
- Name of the action (Research and Innovation Action)
- Acronym and grant number (AUTOPILOT, 731993)
- Publication date
- Length of embargo period if applicable
- Persistent identifier

When referencing Open access data, AUTOPILOT will include at a minimum the following statement demonstrating EU support (with relevant in-formation included into the repository metadata):
- "This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 731993".

The AUTOPILOT consortium will strive to make many of the collected datasets open access. When this is not the case, the data sharing section for that particular dataset will describe why access has been restricted. (See. Chapter 5)

In regards to the specific repositories available to the AUTOPILOT consortium, numerous project partners maintain institutional repositories that is listed in the following DMP version, where project scientific publications and in some instances, research data is deposited. The use of a specific repository will depend primarily on the primary creator of the publication and on the data in question.

Some other project partners will not operate publically accessible institutional repositories. When depositing scientific publications they shall use either a domain specific repository or use the EU recommended service OpenAIRE (http://www.openaire.eu) as an initial step to finding resources to determine relevant repositories.

Project research data shall be deposited to the online data repository ZENODO[16]. It is a free service developed by CERN under the EU FP7 project OpenAIREplus (grant agreement no.283595).
The repository shall also include information regarding the software, tools and instruments

14 http://ec.europa.eu/research/participants/docs/h2020-funding-guide/cross-cutting-issues/open-access-dissemination_en.htm
15 http://ec.europa.eu/research/participants/docs/h2020-funding-guide/cross-cutting-issues/open-access-data-management/open-access_en.htm
16 https://zenodo.org/

that were used by the dataset creator(s) so that secondary data users can access and then validate the results.

The AUTOPILOT data collection can be accessed in ZENODO repository at the following address: https://zenodo.org/communities/autopilot

Note that publishing of data into Zenodo is made by data producers during the project and after the end of the project.

In summary, as a baseline AUTOPILOT partners shall deposit:

- Scientific publications – on their respective institute repositories in addition (when relevant) to the AUTOPILOT Zenodo repository
- Research data – to the AUTOPILOT Zenodo collection (when possible)
- Other project output files – to the AUTOPILOT Zenodo collection (when relevant)

5. AUTOPILOT dataset description

5.1 General Description

This section provides an explanation of the different types of data sets produced or collected in AUTOPILOT project.

The descriptions of the different data sets, including their reference, file format, standards, methodologies and metadata and repository to be used are given below. These descriptions are collected using the pilot sites requirements and specifications.

It is important to note that the dataset bellow is produced by each use case in all the pilot sites. The dataset categories are:

- IoT dataset
- Vehicle dataset
- V2X messages dataset
- Surveys dataset

All Pilot Sites have agreed about logged parameters. Therefore, following table describes all the parameters used across Pilot Sites: most are shared, a subset can be specific to a Pilot Site or a Use Case.

5.2 Template used in dataset description

This table is a template used to describe the datasets.

Table 2 – Dataset description template

Dataset Reference	**AUTOPILOT_PS_UC_datatype_ID** Each data set will have a reference that is generated by the combination of the name of the project, the pilot site (PS) and the use case in which it is generated. **Example**: AUTOPILOT_Versailles_Platooning_IOT_01
Dataset Nature	Nature of the data set
Dataset Description	Each data set will have a full data description explaining the data provenance, origin and usefulness. Reference may be made to existing data that could be reused.
Standards and metadata	The metadata attributes list and standards. The used methodologies
File format	All the format that defines data
Data Sharing	Explanation of the sharing policies related to the data set between the next options: **Open**: Open for public disposal **Embargo**: It iscome public when the embargo period applied by the publisher is over. In case it is categorized as embargo the end date of the embargo period must be written in DD/MM/YYYY format. **Restricted**: Only for project internal use. Each data set must have its distribution license. Provide information about personal data and mention if the

	data is anonymized or not. Tell if the dataset entails personal data and how this issue is taken into account.
Archiving and Preservation	The preservation guarantee and the data storage during and after the project **Example**: databases, institutional repositories, public repositories.

5.3 IOT dataset

Table 3 – IOT dataset description

Dataset Reference	**AUTOPILOT_PS_UC_IOT_ID**
Dataset Nature	IOT data generated from connected devices
Dataset Description	This dataset refer to the IOT datasets that is generated from IOT devices within use cases. This includes the data coming from VRUs, RSUs, smartphones, Vehicles, drones …
Standards and metadata	During the project, The metadata related to the IOT data are based on OneM2M standard. The OneM2M IOT platforms are implemented across pilot sites to cover the interoperability feature. More details are provided in the section 2.2.2. In addition, the data model of these data is inspired from the DMAG (data management activity group) work done in T2.3. The DMAG defined a unified data model that standardizes all the IOT messages across pilot sites. The AUTOPILOT common IOT data model is based on different standards: SENSORIS, DATEX II. After the project, The metadata is enriched with ZENODO's metadata, including the title, creator, date, contributor, pilot site, use case, description, keywords, format, resource type, etc…
File format	JSON, CSV
Data Sharing	This dataset is widely open to be used by 3rd party applications and is deposited in the ZENODO repository.
Archiving and Preservation	During the project, the data will first be stored in the IOT platform. Then, the data is transferred to the pilot site test server before finishing up in the centralized test server. At the end of the project, the data set is archived and preserved in ZENODO repositories.

5.3.1 IOT parameters description

Table 4 – IoT parameters description

Name	**Type**	**Range**	**Unit**	**Description**
rowid	serial	0..	[N/A]	sequence of row numbers to uniquely identify a log line by <log_stationid, log_timestamp, rowid>, only necessary when a subtable is logged
log_timestamp	long	from 0 to 4398046511103	msec	timestamp at which the log_stationid logs (writes) the data row. elapsed

		(= 2^{42}-1)		time since midnight January 1st 1970 UTC
log_stationid	long	from 0 to 4294967295 (= 2^{32}-1)	[N/A]	unique id of the host (e.g. stationid, server id, IoT platform or device id, cloud service id, ...) that logs this log data row. Log_stationid can be another host than the source generating the data to be logged
log_action	enum	['SENT', 'RECEIVED']	[N/A]	Action that triggered the logging event. (Enum: 'SENT', 'RECEIVED')
log_communicationprofile	enum	['ITS_G5', 'CELLULAR', 'UWB', 'LTE_V2X']	[N/A]	Communication profile, medium or path used to send or receive the message. This needs to be logged in case messages are communicated via alternative profiles. Default is ITS_G5. multiple channels are used to communicate similar messages
log_messagetype	enum	['ETSI.CAM', 'ETSI.DENM', 'ISO.IVI', 'ETSI.MAPEM', 'ETSI.SPATEM'] 'IoT.IOT-OBJECT'	[N/A]	Type of standardised message, used for automated processing in case multiple message types are combined in a single log file. The enum fields refer to the <standardisation organisation>.<message type>.
data	string		[N/A]	JSON containing data extracted from IoT platform, according to Use Case.

5.4 Vehicles dataset

Table 5 – Vehicles dataset description

Dataset Reference	**AUTOPILOT_PS_UC_VEHICLES_ID**
Dataset Nature	Data generated from the vehicle sensors.
Dataset Description	This dataset refer to the vehicle datasets that is generated from the vehicle sensors within use cases. This includes the data coming from the CAN bus, cameras, RADARs, LIDARs and GPS. Following log types can be produced by vehicles : - Vehicle - Positioning system - Vehicle dynamics - Driver vehicle interaction - Environment sensors
Standards and metadata	The vehicle data standards used in AUTOPILOT are developed in the task 2.1. The pilot site implementations are based on well-known standards to come up with a common data format: CAN, ROS … More details are provided in D2.1. After the project, The metadata is based on ZENODO's metadata, including the title, creator, date, contributor, pilot

	site, use case, description, keywords, format, resource type, etc…
File format	XML, CSV, SQL, JSON
Data Sharing	This dataset is widely open to be used by 3rd party applications and is deposited in the ZENODO repository.
Archiving and Preservation	During the project, the data will first be stored in pilot site test server before finishing up in the centralized test server. At the end of the project, the data set is archived and preserved in ZENODO repositories.

5.4.1 Vehicles parameters description

All Pilot Sites have agreed about logged parameters. Therefore, following table describes all the parameters used across Pilot Sites: most are common; a subset can be specific to a Pilot Site or a Use Case.

First table describes the common metadata for the data that is logged with every parameter

Table 6 – Vehicle common metadata description

Name	Type	Range	Unit	Description
rowid	serial	0..	[N/A]	sequence of row numbers to uniquely identify a log line by <log_stationid, log_timestamp, rowid>, only necessary when a subtable is logged
log_timestamp	long	from 0 to 4398046511103 (= 2^{42}-1)	msec	timestamp at which the log_stationid logs (writes) the data row. elapsed time since midnight January 1st 1970 UTC
log_stationid	long	from 0 to 4294967295 (= 2^{32}-1)	[N/A]	unique id of the host (e.g. stationid, server id, IoT platform or device id, cloud service id, ...) that logs this log data row. Log_stationid can be another host than the source generating the data to be logged
log_applicationid	long	from 0 to 4294967295 (= 2^{32}-1)	[N/A]	unique id of the application, instance or thread, on the log_stationid host that logs this log data row. Applicationid is at least unique within the log_station. ApplicationId is mandatory if multiple components on a host log to the same table or if the application logging into a table is not trivial (e.g. it is trivial that a CAM Basic Service is the only application logging CAM messages in the cam table). For vehicle data, the log_applicationid is also used to identify specific physical and virtual sensors, such as front_camera, radar, lidar, GPS, CAN

Table 7 – Vehicle parameters description

Name	Type	Range	Unit	Description
speed	double	from 0 to 163.82	[m/s]	Speed over ground, meters per second.

outsidetemperature	double	from -60 to 67	[°C]	Vehicle outside temperature during trip.
insidetemperature	double	from -60 to 67	[°C]	Vehicle inside temperature during trip.
batterysoc	double	from 0 to 100	[%]	Percentage of the battery of the vehicle.
rangeestimated	double	from 0 to 1000	[km]	Range estimated with the actual percentage of the battery and/or available fuel.
fuelconsumption	double	from 0 to 1	[L/km]	Average fuel consumption during a route or trip.
enginespeed	int	from 0 to 10000	[1/min]	Engine speed calculated in terms of revolutions per minute.
owndistance	double	from 0 to 5000	[km]	Total kilometrage per day or trip or road type etc.

Table 8 – Positioning system parameters description

Name	Type	Range	Unit	Description
speed	double	from 0 to 163.82	[m/s]	Speed over ground, meters per second. Measured by GNSS receiver.
longitude	double	from -90 to 90	[degree]	Longitude
latitude	double	from -180 to 180	[degree]	Latitude
heading	double	from 0 to 360	[degree]	Heading
ggasentence	string		[NMEA format]	GGA - Fix information.
gsasentence	string		[NMEA format]	GSA - Overall Satellite data.
rmcsentence	string		[NMEA format]	RMC - recommended minimum data for gps.
vtgsentence	string		[NMEA format]	VTG - Vector track an Speed over the Ground.
zdasentence	string		[NMEA format]	ZDA - Date and Time.

Table 9 – Vehicle dynamics description

Name	Type	Range	Unit	Description
yawrate	double	from -327.66 to 327.66	[°/s]	Vehicle rotation around the centre of mass of the empty vehicle. The leading sign denotes the direction of rotation. The value is negative if the motion is clockwise when viewing from the top.
acclateral	double	from -16 to 16	[m/s2]	Lateral acceleration of the vehicle.
acclongitudinal	double	from -16 to 16	[m/s2]	Longitudinal acceleration of the vehicle.

accvertical	double	from -16 to 16	[m/s2]	Vertical acceleration of the vehicle.
speedwheelunitdistance	double	from 0 to 163.82	[m/s]	Sensor on free running wheel for increased accuracy. Speed measured from wheels (???).
lanechange	enum	['NO' 'YES']	[N/A]	Lane change detection.
speedlimit	int	from 0 to 150	[km/h]	Maximum legal speed limit (log_applicationid identifies the source: updated in real time, from map information, from traffic sign).

Table 10 – Driver vehicle interaction

Name	Type	Range	Unit	Description
throttlestatus	int	from 0 to 100	[%]	Position of the throttle pedal (% pushed). Modify to boolean (i.e., 0->NOT PUSHED, 1-> PUSHED) if % is not available on the car.
clutchstatus	int	from 0 to 100	[%]	Position of the clutch pedal (% pushed). Modify to boolean (i.e., 0->NOT PUSHED, 1-> PUSHED) if % is not available on the car.
brakestatus	int	from 0 to 100	[%]	Position of the brake pedal (% pushed). Modify to boolean (i.e., 0->NOT PUSHED, 1-> PUSHED) if % is not available on the car.
brakeforce	double	from 0 to 300	[bar]	Measure of master cylinder pressure.
wipersstatus	enum	['OFF' 'ON']	[N/A]	Position of the windscreen wipers (boolean). Extend the enumeration if more details are available (e.g., ['OFF', 'SLOW', 'FAST'], ['OFF', 'SLOW1', 'SLOW2', 'FAST1', 'FAST2']).
steeringwheel	double	from -720 to 720	[°]	Position of the steering wheel.
adaptivecruise controlstate	enum	['OFF' 'ON']	[N/A]	ACC activated (ON) / or not (OFF)
adaptivecruise controlsetspeed	double	>0.0	[m/s]	Speed target setting of ACC

Table 11 – Environment sensors absolute

Name	Type	Range	Unit	Description
longitude	double	from -90 to 90	[degree]	Main object transformed to geolocalized coordinates longitudinal (log_applicationid identifies the sensor providing this measurement (e.g., camea, lidar, radar,...)).
latitude	double	from -180 to 180	[degree]	Main object transformed to geolocalized coordinates lateral position (log_applicationid identifies the sensor providing this measurement (e.g., camea, lidar, radar,...)).
obstacle_ID	int	from 0 to 1000	[-]	ID of the obstacle detected by

				environmental sensors
obstacle_covariance	float 64			Covariance matrix of positions of lon, lat, altitude of RADAR detected objects
ObjectClass	int	from 0 to 65	[-]	65 classes from Mapillary dataset --> see http://research.mapillary.com/publication/iccv17a/
lanewidthsensorbased	double	from 0 to 10	[m]	Lane width measured by on-board sensor(s).
lanewidthmapbased	double	from 0 to 10	[m]	Lane width from map information.
trafficsigndescription	string		[N/A]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1
speedlimit_sign	double	from 0 to 250	[km/h]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1
servicecategory	enum	['dangerWarning', 'regulatory', 'informative', 'publicFacilities', 'ambientCondition', 'roadCondition']	[N/A]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1
servicecategorycode	int	[11, 12, 13, 21, 31, 32]	[N/A]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1
countrycode	string		[N/A]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1 (ISO 3166-1 alpha-2)
pictogramcategorycode	int	from 0 to 999	[N/A]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1
VRU_pedestrian_class	int	from 0 - 3	1 = children, 2 = adults, 3 = elderly	sub classes of pedestrians
VRU_cyclist_class	int	from 0 - 3	1 = children, 2 = adults, 3 = elderly	sub classes of cyclists/riders
confidence_levels	double	from 0 - 100	[%]	Indication for false positive detections (minimum default level)
Environ_info	int	from 1 - 6	[-]	1=sunny/day, 2=raining/day, 3=snow/day, 4=night/dry, 5=raining/night, 6=snow/night
Road_hazard	int	from 0 to 42	[N/A]	No standardized dataset available --> current proposal: pothole detection, slippery road, black ice etc.
sensor_position	int	from 0 to 1000	[mm]	position of sensor on vehicle wrt. CoG. required for correlating to environmental

				detetction with lot detections
process_delay	int	from 0 to 1000	[ms]	is processing delay known or unknown?

Table 12 – Environment sensors relative

Name	Type	Range	Unit	Description
x	double	from 0 to 500	[m]	Main object relative distance longitudinal / x-direction (log_applicationid identifies the sensor providing this measurement (e.g., camea, lidar, radar,...)).
y	double	from -50 to 50	[m]	Main object relative distance lateral / y-direction (log_applicationid identifies the sensor providing this measurement (e.g., camea, lidar, radar,...)).
obstacle_ID	int	from 0 to 1000	[-]	ID of the obstacle detected by environmental sensors
obstacle_covariance	float64			Covariance matrix of positions of lon, lat, altitude of RADAR detected objects
ObjectClass	int	from 0 to 65	[-]	65 classes from Mapillary dataset --> see http://research.mapillary.com/publication/iccv17a/
lanewidthsensorbased	double	from 0 to 10	[m]	Lane width measured by on-board sensor(s).
lanewidthmapbased	double	from 0 to 10	[m]	Lane width from map information.
trafficsigndescription	string		[N/A]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1
speedlimit_sign	double	from 0 to 250	[km/h]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1
servicecategory	enum	['dangerWarning', 'regulatory', 'informative', 'publicFacilities', 'ambientCondition', 'roadCondition']	[N/A]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1
servicecategorycode	int	[11, 12, 13, 21, 31, 32]	[N/A]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1
countrycode	string		[N/A]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1 (ISO 3166-1 alpha-2)
pictogramcategorycode	int	from 0 to 999	[N/A]	signrecognition -- as defined in IVI - ISO TS 19321 (2015) v1
VRU_pedestrian_class	int	from 0 - 3	1 = children, 2 = adults, 3 = elderly	sub classes of pedestrians

VRU_cyclist_class	int	from 0 - 3	1 = children, 2 = adults, 3 = elderly	sub classes of cyclists/riders
confidence_levels	double	from 0 - 100	[%]	Indication for false positive detections (minimum default level)
Environ_info	int	from 1 - 6	[-]	1=sunny/day, 2=raining/day, 3=snow/day, 4=night/dry, 5=raining/night, 6=snow/night
Road_hazard	int	from 0 to 42	[N/A]	No standardized dataset available --> current proposal: pothole detection, slippery road, black ice etc.
sensor_position	int	from 0 to 1000	[mm]	position of sensor on vehicle wrt. CoG. required for correlating to environmental detetction with Iot detections
process_delay	int	from 0 to 1000	[ms]	is processing delay known or unknown?

5.5 V2X messages dataset

Table 13 – V2X messages dataset description

Dataset Reference	**AUTOPILOT_PS_UC_V2X_ID**
Dataset Nature	V2X messages communicated during test sessions
Dataset Description	This dataset refer to the V2X messages that is generated from the communication between the vehicles and any other party that could affect the vehicle. This includes the other vehicles and the pilot site infrastructure.
Standards and metadata	The V2X messages are mainly generated from the communication standard ITS-G5. After the project, The metadata is enriched by ZENODO's metadata, including the title, creator, date, contributor, pilot site, use case, description, keywords, format, resource type, etc...
File format	CAM, DEMN, IVI, SPAT, MAP, CSV
Data Sharing	This dataset is widely open to be used by 3rd party applications and is deposited in the ZENODO repository.
Archiving and Preservation	During the project, the data will first be stored in pilot site test server before finishing up in the centralized test server. At the end of the project, the data set is archived and preserved in ZENODO repositories.

5.5.1 V2X parameters description

Table 14 – V2X parameters description

Name	Type	Range	Unit	Description
rowid	serial	0..	[N/A]	sequence of row numbers to uniquely identify a log line by <log_stationid, log_timestamp, rowid>, only

				necessary when a subtable is logged
log_timestamp	long	from 0 to 4398046511103 (= 2^{42}-1)	msec	timestamp at which the log_stationid logs (writes) the data row. elapsed time since midnight January 1st 1970 UTC
log_stationid	long	from 0 to 4294967295 (= 2^{32}-1)	[N/A]	unique id of the host (e.g. stationid, server id, IoT platform or device id, cloud service id, ...) that logs this log data row. Log_stationid can be another host than the source generating the data to be logged
log_applicationid	long	from 0 to 4294967295 (= 2^{32}-1)	[N/A]	unique id of the application, instance or thread, on the log_stationid host that logs this log data row. Applicationid is at least unique within the log_station. ApplicationId is mandatory if multiple components on a host log to the same table or if the application logging into a table is not trivial (e.g. it is trivial that a CAM Basic Service is the only application logging CAM messages in the cam table). For vehicle data, the log_applicationid is also used to identify specific physical and virtual sensors, such as front_camera, radar, lidar, GPS, CAN
log_action	enum	['SENT', 'RECEIVED']	[N/A]	Action that triggered the logging event. (Enum: 'SENT', 'RECEIVED')
log_communicationprofile	enum	['ITS_G5', 'CELLULAR', 'UWB', 'LTE_V2X']	[N/A]	Communication profile, medium or path used to send or receive the message. This needs to be logged in case messages are communicated via alternative profiles. Default is ITS_G5. multiple channels are used to communicate similar messages
log_messagetype	enum	['ETSI.CAM', 'ETSI.DENM', 'ISO.IVI', 'ETSI.MAPEM', 'ETSI.SPATEM']	[N/A]	Type of standardised message, used for automated processing in case multiple message types are combined in a single log file. The enum fields refer to the <standardisation organisation>.<message type>.
log_messageuuid	uuid		[N/A]	Universal Unique Identifier of the message. This is an alternative for the identification of messages from the message contents. If used, then the uuid should also be included in the payload of the message and communicated between senders and receivers.

data	string		[N/A]	CAM or DENM message payload

5.6 Application messages dataset

Table 15 – Applications dataset description

Dataset Reference	**AUTOPILOT_PS_UC_APPLICATION_ID**
Dataset Nature	Application data collected during test sessions
Dataset Description	This data refers to the data generated by AD applications.
Standards and metadata	The application messages are mainly text application status exchanged between AD applications and services. After the project, The metadata is enriched by ZENODO's metadata, including the title, creator, date, contributor, pilot site, use case, description, keywords, format, resource type, etc…
File format	CSV
Data Sharing	This dataset is widely open to be used by 3rd party applications and is deposited in the ZENODO repository.
Archiving and Preservation	During the project, the data will first be stored in pilot site test server before finishing up in the centralized test server. At the end of the project, the data set is archived and preserved in ZENODO repositories.

5.6.1 Application messages parameters description

Table 16 – Platooning event parameters description

Name	**Type**	**Range**	**Unit**	**Description**
rowid	serial	0..	[N/A]	sequence of row numbers to uniquely identify a log line by <log_stationid, log_timestamp, rowid>
log_timestamp	long	from 0 to 4398046511103 (= 2^{42}-1)	[millisecond]	timestamp at which the log_stationid logs (writes) the data row. elapsed time since midnight January 1st 1970 UTC
log_stationid	long	from 0 to 4294967295 (= 2^{32}-1)	[N/A]	unique id of the host (stationid or processing unit id) that logs this log data row. Log_stationid can be another host than the source generating the data to be logged
log_applicationid	long	from 0 to 4294967295 (= 2^{32}-1)	[N/A]	unique id of the application, instance or thread, on the log_stationid host that logs this log data row. Applicationid is at least unique within the log_station. ApplicationId is mandatory if multiple components on a host log

				to the same table or if the application logging into a table is not trivial (e.g. it is trivial that a CAM Basic Service is the only application logging CAM messages in the cam table).
eventtypeid	enum			event type is defined in the EventModels tables, e.g. for GLOSA
eventid	long			unique id defined by the stationid. Note that this cannot be specific to an application within a station
action	enum	['VEHICLE', 'PLATOON_SERVICE']	[N/A]	Action that triggered the logging event. (Enum: 'SENT', 'RECEIVED')

Table 17 – Platooning action parameters description

Name	Type	Range	Unit	Description
rowid	serial	0..	[N/A]	sequence of row numbers to uniquely identify a log line by <log_stationid, log_timestamp, rowid>
log_timestamp	long	from 0 to 4398046511103 (= 2^{42}-1)	[millisecond]	timestamp at which the log_stationid logs (writes) the data row. elapsed time since midnight January 1st 1970 UTC
log_stationid	long	from 0 to 4294967295 (= 2^{32}-1)	[N/A]	unique id of the host (stationid or processing unit id) that logs this log data row. Log_stationid can be another host than the source generating the data to be logged
log_applicationid	long	from 0 to 4294967295 (= 2^{32}-1)	[N/A]	unique id of the application, instance or thread, on the log_stationid host that logs this log data row. Applicationid is at least unique within the log_station. ApplicationId is mandatory if multiple components on a host log to the same table or if the application logging into a table is not trivial (e.g. it is trivial that a CAM Basic Service is the only application logging CAM messages in the cam table).
eventid	long			unique id defined by the

				stationid. Note that this cannot be specific to an application within a station
eventmodelid	int			unique id of the event model from the EventModels sheet
eventactionid	int			unique id of the action from the EventModels sheet
platooningserviceid	?			PlatooningService to subscribe to
leaderstationid	long	from 0 to 4294967295 (= 2^{32}-1)	[N/A]	stationid from platoon leader, e.g. from PlatoonFormation
platoonid	long	from 0 to 4294967295 (= 2^{32}-1)	[N/A]	platoonid e.g. from PlatoonFormation message
generationtime stamputc	long	from 0 to 9223372036854775807 or from 0 to 4398046511103 (= 2^{42}-1) [millisecond]	[millisecond]	generationtimestampITC from PlatoonFormation message

5.7 Surveys dataset

Table 18 – Surveys dataset description

Dataset Reference	**AUTOPILOT_PS_UC_SURVEYS_ID**
Dataset Nature	Surveys data collected during test sessions
Dataset Description	This data refers to the data resulting from the answers of surveys and questionnaires for user acceptance evaluation.
Standards and metadata	Surveys data will use some well-known tools (Google Forms, Survey Monkey …) The work of definition a common format for surveys data is still in progress by the user acceptance evaluation team. After the project, The metadata is enriched by ZENODO's metadata, including the title, creator, date, contributor, pilot site, use case, description, keywords, format, resource type, etc…
File format	CSV, PDF, XLS
Data Sharing	This dataset is widely open to be used by 3rd party applications and is deposited in the ZENODO repository. It is important to note that these data is **anonymized** before data sharing.
Archiving and Preservation	During the project, the data will first be stored in pilot site test server before finishing up in the centralized test server. At the end of the project, the data set is archived and preserved in ZENODO repositories.

5.8 Brainport datasets

5.8.1 Platooning

Table 19 – Brainport platooning datasets description

AUTOPILOT_BrainPort_Platooning_DriverVehicleInteraction	
Dataset Nature	Data extracted from the CAN of the vehicle
Dataset Description	This dataset contains e.g. throttlestatus, clutchstatus, brakestatus, brakeforce, wipersstatus, steeringwheel for the vehicle
File format	CSV
AUTOPILOT_BrainPort_Platooning_EnvironmentSensorsAbsolute	
Dataset Nature	Data extracted from the vehicle environment sensors
Dataset Description	This dataset contains information about detected object, with absolute coordinates
File format	CSV
AUTOPILOT_BrainPort_Platooning_EnvironmentSensorsRelative	
Dataset Nature	Data extracted from the vehicle environment sensors
Dataset Description	This dataset contains information about detected object, with relative coordinates
File format	CSV
AUTOPILOT_BrainPort_Platooning_IotVehicleMessage	
Dataset Nature	Data sent between all devices, vehicles and services
Dataset Description	Each sensor data submission is a Message. A Message has an Envelope, a Path, and optionally (but likely) Path Events and optionally Path Media. The envelope bears fundamental information about the individual sender (the vehicle) but not to a level that owner of the vehicle can be identified or different messages can be identified that originate from a single vehicle.
File format	CSV
AUTOPILOT_BrainPort_Platooning_PlatoonFormation	
Dataset Nature	Data sent from PlatoonService to vehicle
Dataset Description	This dataset contains information about the route and speed for a specific vehicle for forming a platoon
File format	CSV
AUTOPILOT_BrainPort_Platooning_PlatooningAction	
Dataset Nature	Data logged by vehicle
Dataset Description	This dataset contains information about the current status of the platooning
File format	CSV
AUTOPILOT_BrainPort_Platooning_PlatooningEvent	
Dataset Nature	Data logged by vehicle
Dataset Description	This dataset contains information about the identifiers used for each specific platooning event

File format	CSV
AUTOPILOT_BrainPort_Platooning_PlatoonStatus	
Dataset Nature	Data sent by vehicle to PlatoonService
Dataset Description	This dataset contains information about the current status of the platooning
File format	CSV
AUTOPILOT_BrainPort_Platooning_PositioningSystem	
Dataset Nature	Data from GPS on the vehicle
Dataset Description	This dataset contains speed, longitude, latitude, heading from the GPS
File format	CSV
AUTOPILOT_BrainPort_Platooning_PositioningSystemResample	
Dataset Nature	Data from GPS on the vehicle
Dataset Description	This dataset contains speed,longitude,latitude,heading from the GPS, resampled to 100 milliseconds
File format	CSV
AUTOPILOT_BrainPort_Platooning_PSInfo	
Dataset Nature	Data sent by PlatoonService to the vehicle
Dataset Description	This dataset contains speed and route information for the vehicle to create a platoon
File format	CSV
AUTOPILOT_BrainPort_Platooning_Target	
Dataset Nature	Data from sensors on the vehicle
Dataset Description	Target detection in the vicinity of the host vehicle, by a vehicle sensor or virtual sensor
File format	CSV
AUTOPILOT_BrainPort_Platooning_Vehicle	
Dataset Nature	Data from the CAN and sensors about the state of the vehicle
Dataset Description	This dataset contains a.o temperature and battery state of the vehicles
File format	CSV
AUTOPILOT_BrainPort_Platooning_VehicleDynamics	
Dataset Nature	Data from the CAN and sensors about the state of the vehicle
Dataset Description	This dataset contains a.o accelerations and speedlimit of the vehicle, as observed from the CAN and the external sensors
File format	CSV
AUTOPILOT_BrainPort_Platooning_VehicleDynamics	
Dataset	Data from the CAN and sensors about the state of the vehicle

Nature	
Dataset Description	This dataset contains a.o accelerations and speedlimit of the vehicle, as observed from the CAN and the external sensors
File format	CSV

5.8.2 Automated valet parking

Table 20 – Brainport automated valet parking datasets description

AUTOPILOT_BrainPort_AutomatedValetParking_DriverVehicleInteraction	
Dataset Nature	Data extracted from the CAN of the vehicle
Dataset Description	This dataset contains e.g. throttlestatus, clutchstatus, brakestatus, brakeforce, wipersstatus, steeringwheel for the vehicle
File format	CSV
AUTOPILOT_BrainPort_AutomatedValetParking_DroneAvpCommand	
Dataset Nature	Data sent from drone
Dataset Description	This dataset contains route information for a vehicle to a designated parking spot
File format	CSV
AUTOPILOT_BrainPort_AutomatedValetParking_EnvironmentSensorsAbsolute	
Dataset Nature	Data extracted from the vehicle environment sensors
Dataset Description	This dataset contains information about detected object, with absolute coordinates
File format	CSV
AUTOPILOT_BrainPort_AutomatedValetParking_EnvironmentSensorsRelative	
Dataset Nature	Data extracted from the vehicle environment sensors
Dataset Description	This dataset contains information about detected object, with relative coordinates
File format	CSV
AUTOPILOT_BrainPort_AutomatedValetParking_IotVehicleMessage	
Dataset Nature	Data sent between all devices, vehicles and services
Dataset Description	Each sensor data submission is a Message. A Message has an Envelope, a Path, and optionally (but likely) Path Events and optionally Path Media. The envelope bears fundamental information about the individual sender (the vehicle) but not to a level that owner of the vehicle can be identified or different messages can be identified that originate from a single vehicle.
File format	CSV
AUTOPILOT_BrainPort_AutomatedValetParking_ParkingSpotDetection	
Dataset Nature	Data sent from drone to parkingService
Dataset Description	This dataset contains informaton about detected parking spots

File format	CSV
AUTOPILOT_BrainPort_AutomatedValetParking_PositioningSystem	
Dataset Nature	Data from GPS on the vehicle
Dataset Description	This dataset contains speed, longitude, latitude, heading from the GPS
File format	CSV
AUTOPILOT_BrainPort_AutomatedValetParking_PositioningSystemResampled	
Dataset Nature	Data from GPS on the vehicle
Dataset Description	This dataset contains speed,longitude,latitude,heading from the GPS, resampled to 100 milliseconds
File format	CSV
AUTOPILOT_BrainPort_AutomatedValetParking_Vehicle	
Dataset Nature	Data from the CAN and sensors about the state of the vehicle
Dataset Description	This dataset contains a.o temperature and battery state of the vehicles
File format	CSV
AUTOPILOT_BrainPort_AutomatedValetParking_VehicleAvpCommand	
Dataset Nature	Data sent from ParkingService to vehicle
Dataset Description	This dataset contains route to parkingspot, and some other environmental information
File format	CSV
AUTOPILOT_BrainPort_AutomatedValetParking_VehicleAvpStatus	
Dataset Nature	Data sent from vehicle to ParkingService
Dataset Description	This dataset contains information about the current status and parkingstatus of the vehicle
File format	CSV
AUTOPILOT_BrainPort_AutomatedValetParking_VehicleDynamics	
Dataset Nature	Data from the CAN and sensors about the state of the vehicle
Dataset Description	This dataset contains a.o accelerations and speedlimit of the vehicle, as observed from the CAN and the external sensors
File format	CSV

5.8.3 Highway pilot

Table 21 – Brainport highway pilot datasets description

AUTOPILOT_BrainPort_HighwayPilot_AdasCommand	
Dataset Nature	Data from the automated driver assistance system
Dataset	This dataset contains the ADAS command in the vehicle

Description	
File format	CSV
AUTOPILOT_BrainPort_HighwayPilot_Anomaly	
Dataset Nature	Data sent from detecting vehicle to the service, and from service to vehicles
Dataset Description	This dataset contains information about all the detected anomalies on the road
File format	CSV
AUTOPILOT_BrainPort_HighwayPilot_AnomalyImage	
Dataset Nature	Data sent from detecting vehicle to service
Dataset Description	This dataset contains images of the detected anomalies
File format	CSV
AUTOPILOT_BrainPort_HighwayPilot_DriverVehicleInteraction	
Dataset Nature	Data extracted from the CAN of the vehicle
Dataset Description	This dataset contains e.g. throttlestatus,clutchstatus,brakestatus,brakeforce,wipersstatus,steeringwheel for the vehicle
File format	CSV
AUTOPILOT_BrainPort_HighwayPilot_EnvironmentSensorsAbsolute	
Dataset Nature	Data extracted from the vehicle environment sensors
Dataset Description	This dataset contains information about detected object, with absolute coordinates
File format	CSV
AUTOPILOT_BrainPort_HighwayPilot_EnvironmentSensorsRelative	
Dataset Nature	Data extracted from the vehicle environment sensors
Dataset Description	This dataset contains information about detected object, with relative coordinates
File format	CSV
AUTOPILOT_BrainPort_HighwayPilot_Hazard	
Dataset Nature	Data sent from service to vehicle
Dataset Description	This dataset contains specific information for a vehicle about anomalies and hazards
File format	CSV
AUTOPILOT_BrainPort_HighwayPilot_IotVehicleMessage	
Dataset Nature	Data sent between all devices, vehicles and services
Dataset Description	Each sensor data submission is a Message. A Message has an Envelope, a Path, and optionally (but likely) Path Events and optionally Path Media. The envelope bears fundamental information about the individual sender (the

	vehicle) but not to a level that owner of the vehicle can be identified or different messages can be identified that originate from a single vehicle.
File format	CSV
AUTOPILOT_BrainPort_HighwayPilot_PositioningSystem	
Dataset Nature	Data from GPS on the vehicle
Dataset Description	This dataset contains speed, longitude, latitude, heading from the GPS
File format	CSV
AUTOPILOT_BrainPort_HighwayPilot_Vehicle	
Dataset Nature	Data from the CAN and sensors about the state of the vehicle
Dataset Description	This dataset contains a.o temperature and battery state of the vehicles
File format	CSV
AUTOPILOT_BrainPort_HighwayPilot_VehicleDynamics	
Dataset Nature	Data from the CAN and sensors about the state of the vehicle
Dataset Description	This dataset contains a.o accelerations and speedlimit of the vehicle, as observed from the CAN and the external sensors
File format	CSV

5.8.4 Urban driving

Table 22 – Brainport urban driving datasets description

AUTOPILOT_BrainPort_ UrbanDriving_DriverVehicleInteraction	
Dataset Nature	Data extracted from the CAN of the vehicle
Dataset Description	This dataset contains e.g. throttlestatus, clutchstatus, brakestatus, brakeforce, wipersstatus, steeringwheel for the vehicle
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_EAI2Mobile	
Dataset Nature	Data from the service to the mobile
Dataset Description	This dataset contains information sent to the mobile about the Estimated Arrival time and position
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_EnvironmentSensorsAbsolute	
Dataset Nature	Data extracted from the vehicle environment sensors
Dataset Description	This dataset contains information about detected object, with absolute coordinates
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_EnvironmentSensorsRelative	
Dataset	Data extracted from the vehicle environment sensors

Nature	
Dataset Description	This dataset contains information about detected object, with relative coordinates
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_IOT_CEMA_Message	
Dataset Nature	Data from the service to the vehicle
Dataset Description	This dataset contains information from the Crowd Estimation and Mobility Analytics service
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_IOT_FlowRadar_Message	
Dataset Nature	Data from the vehicle to the service
Dataset Description	This dataset contains the GPS informaton (speed,position,heading) from the vehicle
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_IotVehicleMessage	
Dataset Nature	Data sent between all devices, vehicles and services
Dataset Description	Each sensor data submission is a Message. A Message has an Envelope, a Path, and optionally (but likely) Path Events and optionally Path Media. The envelope bears fundamental information about the individual sender (the vehicle) but not to a level that owner of the vehicle can be identified or different messages can be identified that originate from a single vehicle.
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_IOT_VehicleStatus	
Dataset Nature	Data sent from the vehicle to the service
Dataset Description	This dataset contains the current status of the vehicle
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_PositioningSystem	
Dataset Nature	Data from GPS on the vehicle
Dataset Description	This dataset contains speed,longitude,latitude,heading from the GPS
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_SmartphoneGPS	
Dataset Nature	Data sent by the mobile to the service
Dataset Description	This dataset contains the GPS informaton (speed,position,heading) from the mobile
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_SmartphoneStatus	
Dataset Nature	Data sent from the mobile to the service

Dataset Description	This dataset contains the current status of the mobile
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_TaxiRequest	
Dataset Nature	Data sent from the mobile to the service
Dataset Description	This dataset contains the requests for a taxi from the mobile phones
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_Vehicle	
Dataset Nature	Data from the CAN and sensors about the state of the vehicle
Dataset Description	This dataset contains a.o temperature and battery state of the vehicles
File format	CSV
AUTOPILOT_BrainPort_ UrbanDriving_VehicleDynamics	
Dataset Nature	Data from the CAN and sensors about the state of the vehicle
Dataset Description	This dataset contains a.o accelerations and speedlimit of the vehicle, as observed from the CAN and the external sensors
File format	CSV

5.9 Livorno datasets

5.9.1 Urban driving

Table 23 – Livorno urban driving datasets description

AUTOPILOT_Livorno_UrbanDriving_Vehicle_all	
Dataset Nature	Data generated from the vehicle sensors
Dataset Description	This dataset refers to the vehicle datasets generated from the vehicle sensors during Platooning at Versailles. This includes the data coming from the CAN bus and GPS. It includes following kind of dataset: Vehicle: general data (speed, battery) PositioningSystem: data from GPS VehicleDynamics: data about dynamic (acceleration...) LateralControl: steering and lane control data
File format	CSV
AUTOPILOT_Livorno_UrbanDriving_V2X_all	
Dataset Nature	V2V messages during platooning sessions
Dataset Description	This dataset refers to the V2V messages exchanged between ITS stations (vehicles and RSUs) during the Urban Drining in Livorno.
File format	CSV
AUTOPILOT_Livorno_UrbanDriving_IoT_all	

Dataset Nature	Data extracted from IoT oneM2M platform
Dataset Description	This dataset refers to messages exchanged by Urban Driving devices, applications and services across the oneM2M platform.
File format	CSV

5.9.2 Highway pilot

Table 24 – Livorno highway pilot datasets description

AUTOPILOT_Livorno_HighwayPilot_Vehicle_all	
Dataset Nature	Data generated from the vehicle sensors
Dataset Description	This dataset refers to the vehicle datasets generated from the vehicle sensors during Platooning at Versailles. This includes the data coming from the CAN bus and GPS. It includes following kind of dataset: Vehicle: general data (speed, battery) PositioningSystem: data from GPS VehicleDynamics: data about dynamic (acceleration...) LateralControl: steering and lane control data
File format	CSV
AUTOPILOT_Livorno_HighwayPilot_V2X_all	
Dataset Nature	V2V messages during platooning sessions
Dataset Description	This dataset refers to the V2V messages exchanged between ITS stations (vehicles and RSUs) during the Highway Piloting in Livorno.
File format	CSV
AUTOPILOT_Livorno_HighwayPilot_IoT_all	
Dataset Nature	Data extracted from IoT oneM2M platform
Dataset Description	This dataset refers to messages exchanged by HighwayPilot devices, applications and services across the oneM2M platform.
File format	CSV

5.10 Versailles datasets

5.10.1 Platooning

Table 25 – Versailles platooning datasets description

AUTOPILOT_Versailles_Platooning_Vehicle	
Dataset Nature	Data generated from the vehicle sensors
Dataset Description	Vehicle datasets generated by the vehicle sensors during platooning at Versailles. This includes the data coming from the CAN bus and GPS. It includes following kind of datasets: Vehicle: general data (speed, battery) PositioningSystem: data from GPS

	VehicleDynamics: data about dynamic (acceleration...) LateralControl: steering and lane control data
File format	CSV
AUTOPILOT_Versailles_Platooning_V2X	
Dataset Nature	V2X messages during platooning sessions
Dataset Description	This dataset refers to the V2V messages exchanged between the vehicles during the platooning at Versailles. TCPwlan logs contain mainly the identification of sender and receiver and the payload extracted from TCP messages captured on the CAN bus.
File format	CSV
AUTOPILOT_Versailles_Platooning_IoT	
Dataset Nature	Data extracted from IoT oneM2M platform
Dataset Description	This dataset refers to messages exchanged by platooning application and services, Traffic Light Assist service, traffic light controllers across oneM2M platform.
File format	CSV

5.10.2 Urban driving

Table 26 – Versailles urban driving datasets description

AUTOPILOT_Versailles_UrbanDriving_Vehicle	
Dataset Nature	Data generated from the vehicle sensors
Dataset Description	Vehicle datasets generated by the vehicle sensors during urban driving at Versailles. This includes the data coming from the CAN bus and GPS. It includes following kind of datasets: Vehicle: general data (speed, battery) PositioningSystem: data from GPS VehicleDynamics: data about dynamic (acceleration...) Accel: acceleration data EnvironmentSensorsAbsolute: environment sensors in absolute coordinates
File format	CSV
AUTOPILOT_Versailles_UrbanDriving_V2X	
Dataset Nature	V2X messages during urban driving sessions
Dataset Description	Data exchanged with other vehicles and pedestrian during urban driving. SortOfCam: messages sent or received from bicycles
File format	CSV
AUTOPILOT_Versailles_UrbanDriving_IoT	
Dataset Nature	Data extracted from IoT oneM2M platform
Dataset Description	This dataset refers to messages exchanged by urban driving and car sharing application with vehicle, across oneM2M platform. oneM2M: car sharing status data
File format	CSV

AUTOPILOT_Versailles_UrbanDriving_CAM	
Dataset Nature	CAM messages
Dataset Description	This dataset refers to messages captured inside the vehicle during car sharing.
File format	CSV

5.11 Vigo datasets

5.11.1 Automated valet parking

Table 27 – Vigo automated valet parking datasets description

AUTOPILOT_Vigo_Automated_Valet_Parking_Vehicle_all	
Dataset Nature	Data generated from the vehicle sensors
Dataset Description	Datasets generated from the vehicle sensors
File format	CSV
AUTOPILOT_Vigo_Automated_Valet_Parking_V2X_all	
Dataset Nature	V2X messages during avp sessions
Dataset Description	This dataset refer to the V2X messages generated from the communication between the vehicle and the infrastructure.
File format	CSV
AUTOPILOT_Vigo_Automated_Valet_Parking_IoT_all	
Dataset Nature	Data extracted from IoT oneM2M platform
Dataset Description	This dataset refer to the IOT datasets generated from IOT devices
File format	CSV

5.11.2 Urban driving

Table 28 – Vigo urban driving datasets description

AUTOPILOT_Vigo_Automated_Urban_Driving_Vehicle_all	
Dataset Nature	Data generated from the vehicle sensors
Dataset Description	Datasets generated from the vehicle sensors
File format	CSV
AUTOPILOT_Vigo_Automated_Urban_Driving_V2X_all	
Dataset Nature	V2X messages during urban driving sessions
Dataset Description	This dataset refer to the V2X messages generated from the communication between the vehicle, others vehicles and the infrastructure.

File format	CSV
AUTOPILOT_Vigo_Automated_Urban_Driving_IoT_all	
Dataset Nature	Data extracted from IoT oneM2M platform
Dataset Description	This dataset refer to the IOT datasets generated from IOT devices
File format	CSV

6. FAIR data management principles

The data that is generated during and after the project should be **FAIR**[17], that is Findable, Accessible, Interoperable and Reusable. These requirements do not affect implementation choices and don't necessarily suggest any specific technology, standard, or implementation solution.

The FAIR principles were generated to improve the practices for data management and data-curation, and FAIR aims to describe the principles in order to be applied to a wide range of data management purposes, whether it is data collection or data management of larger research projects regardless of scientific disciplines.

With the endorsement of the FAIR principles by H2020 and their implementation in the guidelines for H2020, The FAIR principles serve as a template for lifecycle data management and ensure that the most important components for lifecycle are covered.

This is intended as an implementation of the FAIR concept rather than a strict technical implementation of the FAIR principles. AUTOPILOT project has implemented several actions described below to carry on the FAIR principles.

Making data findable, including provisions for metadata

- The data sets will have very rich metadata to facilitate the findability. In particular for IOT data, the metadata are based on the OneM2M standard.
- All the data sets will have a Digital Object Identifiers provided by the public repository (ZENODO)
- The reference used for the data set will follow this format : **AUTOPILOT_PS_UC_Datatype_XX**
- The standards for metadata are defined in the chapter 5 tables and explained in section 2.2.

Making data openly accessible

- All the data sets that are openly available are described in the chapter 5.
- The data sets for evaluation is accessible via AUTOPILOT centralized test server.
- The data sets is made available using public repository (e.g. ZENODO) after the project
- The data sharing in chapter 5 explains the methods or software used to access the data. Basically, no software is needed to access the data
- The data and their associated metadata is deposed in a public repository or either in an institutional repository.
- The data sharing in the section 4 will outline the rules to access the data if restrictions exist

Making data interoperable

- The metadata vocabularies, standards and methodologies will depend on the public repository and are mentioned in the chapter 5 0tables.
- The AUTOPILOT WP2 made several actions in order to define common data formats. This work was developed in task 2.1 for vehicle data and task 2.3 for Iot data. The

[17] http://ec.europa.eu/research/participants/docs/h2020-funding-guide/cross-cutting-issues/open-access-data-management/data-management_en.htm

goal is to have the same structure across pilot sites and enable evaluators dealing with the same format for all pilot sites.

- AUTOPILOT PS use IOT platforms based on OneM2M standards to enable data interoperability across pilot sites.

Increase data re-use (through clarifying licenses)

- All the data producers will license their data to allow the widest reuse possible.
- By default, the data is made available for reuse. If any constrains exist, an embargo period is mentioned in the section 4 tables to keep the data for only a period of time
- The data producers will make their data for third-parties within public repositories. They is reused for the scientific publications validation purpose

7. Responsibilities

In order to face the data management challenges efficiently, All AUTOPILOT partners have to respect the policies set out in this DMP and datasets have to be created, managed and stored appropriately.

The Data controller role within AUTOPILOT is undertaken by Francois Fischer (ERTICO) who will directly report to the AUTOPILOT Ethics Board. The Data controller acts as the point of contact for Data Protection issue and will coordinate the actions required to liaise between different beneficiaries and their affiliates, as well as their respective Data Protection agencies, in order to ensure that data collection and processing within the scope of AUTOPILOT, is carried out according to EU and national legislation. Regarding the ORDP, the data controller must ensure that data are shared and easily available.

Each data producer and WPL is responsible for the integrity and compatibility of its data during the project lifetime. The data producer is responsible for sharing its datasets through open access repositories. He is in charge of providing the latest version.

Regarding ethical issues, the deliverable D7.1 details all the measures that AUTOPILOT will use to comply with the H2020 Ethics requirements.

The Data Manager role within AUTOPILOT will directly report to the Technical Meeting Team (TMT). The Data Manager will coordinate the actions related to data management and in particular the compliance to Open Research Data Pilot guideline. The data manager is responsible for implementing the data management plan and he ensures it is reviewed and revised.

8. Ethical issues and legal compliance

As explained in Chapter 2, the central IoT platform is a cloud platform that is hosted on IBM infrastructure, and maintained by IBM IE. It will integrate and aggregate data from the various vehicles and pilot sites.

All data transfers to the IBM hosted central IoT Platform are subject to and conditional upon compliance with the following requirements:

- Prior to any transfer of data to the IBM hosted central IoT platform, all partners must execute an agreement as provided for in Attachment 6 of Autopilot Collaboration Agreement.
- All the partners must agree to commit not to provide personal data to the central IoT platform and to represent that it secures all necessary authorizations & consents before sharing data or any other type of information ("Background, Results, Confidential Information and/or any data") with other parties.
- Every partner that needs to send and store data in the central IoT platform has to request access to the servers, and inform IBM IE what type of data they will send.
- IBM IE will review all data sources BEFORE approving them and allowing them into the central IoT platform, to ensure they are transformed into data that cannot be traced back to personal information.
- No raw videos/images or private information can be sent to the central IoT platform. The partners who will send data to the platform must anonymize data first. Only anonymized information that is extracted from the raw images/videos (e.g., distance between cars, presence of pedestrians, etc.) is accepted and stored.
- The central IoT platform will only be made available to the consortium partners, and not to external entities.
- IBM IE reserves the right to suspend partner's access in case of any suspicious activities detected or non-compliant data received. IBM IE may re-grant access to the platform if a solution demonstrating how to prevent such sharing of personal data and sensitive personal data is reached and implemented.
- IBM IE may implement validation procedures to check that the submitted data structures and types are compliant with what the partners promised to send to the central IoT platform.
- All the data is deleted at the end of the project from all servers of the central IoT platform.

9. Conclusion

This deliverable provides an overview of the data that AUTOPILOT project produced together with related data processes and requirements taken into consideration.

The document outlines an overview about the data set types with detailed description and explains the processes followed for test sites and evaluation within high level representations.

The chapter 5, which describes the data sets, has been enriched with the last progress of the project comparing to the previous version of the DMP (D6.9). This includes detailed description of the standards, methodologies, sharing policies and storage methods.

This final version of the Data Management Plan provides all the details concerning the datasets. These dataset are the results of the test sessions performed at pilot sites level. However, some additional data may be provided later as soon as the partners agree on the data to be shared.

Work on data preparation and data processing are ongoing and will enable help increasing the amount of data to be shared in public repository after the project according to ORDP initiative.

10. Annex

10.1 Annex 1: Open research data pilot

Open access refers to the online provision of scientific information that is free of charge to the end-user and reusable. This scientific information handles the peer-reviewed scientific research articles/publications and the research data underlying publications.

Under the H2020, the project must also aim to deposit the research data needed to validate the results presented in the deposited scientific publications, known as “underlying data”. In order to effectively supply this data, projects need to consider at an early stage how they are going to manage and share the data they create or generate under H2020 guidelines on data management and with respect of AUTOPILOT Grant Agreement.

“The Commission is running a flexible pilot under Horizon 2020 called the **Open Research Data Pilot** (ORD pilot). The ORD pilot aims to improve and maximize access to and re-use of research data generated by Horizon 2020 projects and takes into account the need to balance openness and protection of scientific information, commercialization and Intellectual Property Rights (IPR), privacy concerns, security as well as data management and preservation questions.”

“By extending the pilot, open access becomes the default setting for research data generated in Horizon 2020.
However, not all data can be open. Projects can therefore opt out at any stage (either before or after signing the grant) and so free themselves retroactively from the obligations associated with the conditions – if:

- participation is incompatible with the obligation to protect results that can reasonably be expected to be commercially or industrially exploited
- participation is incompatible with the need for confidentiality in connection with security issues
- participation is incompatible with rules on protecting personal data
- participation would mean that the project's main aim might not be achieved
- The project will not generate / collect any research data or
- There are other legitimate reasons (you can enter these in a free-text box at the proposal stage).”

After depositing publications beneficiaries must ensure open access to those publications via the chosen repository.
“The two main routes to open access are:

- Self-archiving / 'green' open access – the author, or a representative, archives (deposits) the published article or the final peer-reviewed manuscript in an online repository before, at the same time as, or after publication. Some publishers request that open access be granted only after an embargo period has elapsed.

Open access publishing / 'gold' open access - an article is immediately published in open access mode. In this model, the payment of publication costs is shifted away from subscribing readers. “

In the research context, examples of data include statistics, results of experiments, measurements, observations resulting from fieldwork, survey results, interview recordings and images. The focus is on research data that is available in digital format and stored in a public repository. Normally, users can access, mine, exploit, reproduce, and disseminate openly accessible research data free of charge as explained in the following figure.

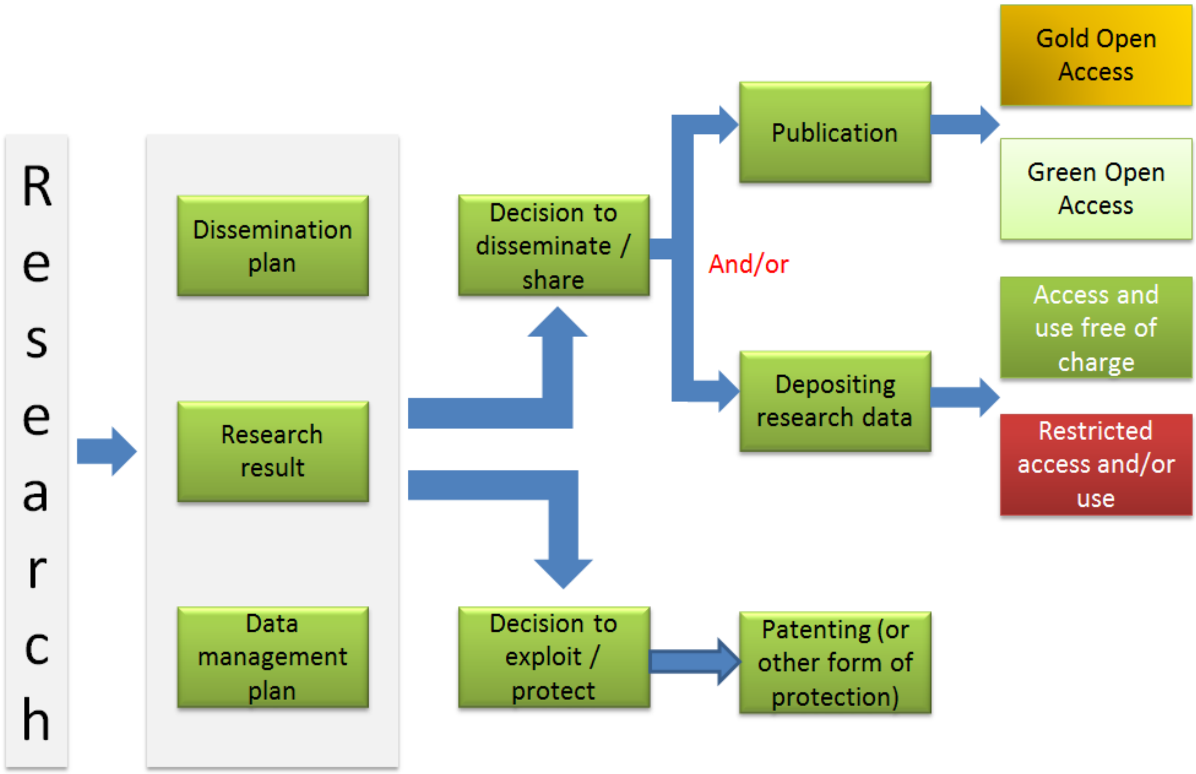

Figure 5 – Principles of H2020 open access to research data

10.2 Annex 2: list of publishable test data from CTS

10.2.1 Brainport, platooning

Applicable datasets references
AUTOPILOT_BrainPort_Platooning_DriverVehicleInteraction, AUTOPILOT_BrainPort_Platooning_EnvironmentSensorsAbsolute, AUTOPILOT_BrainPort_Platooning_EnvironmentSensorsRelative, AUTOPILOT_BrainPort_Platooning_IotVehicleMessage, AUTOPILOT_BrainPort_Platooning_PlatoonFormation, AUTOPILOT_BrainPort_Platooning_PlatooningAction, AUTOPILOT_BrainPort_Platooning_PlatooningEvent, AUTOPILOT_BrainPort_Platooning_PlatoonStatus, AUTOPILOT_BrainPort_Platooning_PositioningSystem, AUTOPILOT_BrainPort_Platooning_PositioningSystemResampled, AUTOPILOT_BrainPort_Platooning_PSInfo, AUTOPILOT_BrainPort_Platooning_Target, AUTOPILOT_BrainPort_Platooning_Vehicle, AUTOPILOT_BrainPort_Platooning_VehicleDynamics

Scenario description	Session description	CTS ID list
Platoon formation and platooning, from Helmond to Eindhoven and back to the Automotive Campus. - Starting in urban area with speed limits of 15 and 30 km/h. - Driving East on the Europaweg with speed limits of 50 and 70 km/h. This includes 3 crossings with traffic lights. - Driving on the the N270, along the Automotive Campus. One crossing with traffic lights, just before the A270. - Driving on the A270 (speed limit 100 km/h). Interrupted by one traffic light. - U-turn at the fly-over or at the end of the A270, to return the same way to the Automotive Campus.	Platoon formation and platooning, without live traffic light data included in planner, but with automatic speed advice. - No live traffic light data available for planner - Driver uses the Android app if Technolution gives a flexible route - Vehicle1 is starting at another location in Helmond (#2), vehicle2 at default location (Automotive Campus) - Platooning (CACC and lane keeping) on the A270 when possible.	204, 205, 206, 207

Platoon formation and platooning, from Helmond to Eindhoven and back to the Automotive Campus. - Starting in urban area with speed limits of 15 and 30 km/h. - Driving East on the Europaweg with speed limits of 50 and 70 km/h. This includes 3 crossings with traffic lights. - Driving on the the N270, along the Automotive Campus. One crossing with traffic lights, just before the A270. - Driving on the A270 (speed limit 100 km/h). Interrupted by one traffic light. - U-turn at the fly-over or at the end of the A270, to return the same way to the Automotive Campus.	Platoon formation and platooning, without live traffic light data included in planner, but with automatic speed advice. - No live traffic light data available for planner - Driver uses the Android app if Technolution gives a flexible route - Starting at default locations - Platooning (CACC and lane keeping) on the A270 when possible.	193
Platoon formation and platooning, from Helmond to Eindhoven and back to the Automotive Campus. - Starting in urban area with speed limits of 15 and 30 km/h. - Driving East on the Europaweg with speed limits of 50 and 70 km/h. This includes 3 crossings with traffic lights. - Driving on the the N270, along the Automotive Campus. One crossing with traffic lights, just before the A270. - Driving on the A270 (speed limit 100 km/h). Interrupted by one traffic light. - U-turn at the fly-over or at the end of the A270, to return the same way to the Automotive Campus.	Baseline of platoon formation only, up to A270, no platooning. Without any IoT, so no speed advices. - No live traffic light data available for planner - Starting at default locations - Meeting at the location of vehicle2 on the Automotive Campus - No platooning or other automated driving.	199
Platoon formation with ONLY AVERAGE traffic light data included in planner. - Disabled live traffic light data included in planner - Planner advices are based on moving average of a certain amount of past traffic light cycles. - Not using the Android app - Starting at default locations	Platoon formation improvement by traffic light data.	83
Driving without speed advice or any services to measure base-line.	Baseline	75, 98, 100

Platoon formation and platooning, from Helmond to Eindhoven and back to the Automotive Campus. - Starting in urban area with speed limits of 15 and 30 km/h. - Driving East on the Europaweg with speed limits of 50 and 70 km/h. This includes 3 crossings with traffic lights. - Driving on the the N270, along the Automotive Campus. One crossing with traffic lights, just before the A270. - Driving on the A270 (speed limit 100 km/h). Interrupted by one traffic light. - U-turn at the fly-over or at the end of the A270, to return the same way to the Automotive Campus.	Baseline of platoon formation and platooning, without any IoT, so no speed advices. - No live traffic light data available for planner - Starting at default locations - Meeting at the location of vehicle2 on the Automotive Campus - Platooning (CACC and lane keeping) on the A270 when possible.	192, 197, 198
Platoon formation and platooning, from Helmond to Eindhoven and back to the Automotive Campus. - Starting in urban area with speed limits of 15 and 30 km/h. - Driving East on the Europaweg with speed limits of 50 and 70 km/h. This includes 3 crossings with traffic lights. - Driving on the the N270, along the Automotive Campus. One crossing with traffic lights, just before the A270. - Driving on the A270 (speed limit 100 km/h). Interrupted by one traffic light. - U-turn at the fly-over or at the end of the A270, to return the same way to the Automotive Campus.	Platoon formation and platooning, without live traffic light data included in planner. - No live traffic light data available for planner - Driver uses the Android app - Starting at default locations - Platooning (CACC and lane keeping) on the A270 when possible.	121, 122, 123, 124, 125, 126
Driving without speed advice or any services to measure base-line.	Baseline (rush hour)	81

Platoon formation and platooning, from Helmond to Eindhoven and back to the Automotive Campus. - Starting in urban area with speed limits of 15 and 30 km/h. - Driving East on the Europaweg with speed limits of 50 and 70 km/h. This includes 3 crossings with traffic lights. - Driving on the the N270, along the Automotive Campus. One crossing with traffic lights, just before the A270. - Driving on the A270 (speed limit 100 km/h). Interrupted by one traffic light. - U-turn at the fly-over or at the end of the A270, to return the same way to the Automotive Campus.	Platoon formation and platooning, without live traffic light data included in planner, but with automatic speed advice. - No live traffic light data available for planner - Driver uses the Android app if Technolution gives a flexible route - Starting at default locations - Platooning (CACC and lane keeping) on the A270 when possible.	194, 195, 196, 200, 201, 202, 203
Platoon formation and platooning, from Helmond to Eindhoven and back to the Automotive Campus. - Starting in urban area with speed limits of 15 and 30 km/h. - Driving East on the Europaweg with speed limits of 50 and 70 km/h. This includes 3 crossings with traffic lights. - Driving on the the N270, along the Automotive Campus. One crossing with traffic lights, just before the A270. - Driving on the A270 (speed limit 100 km/h). Interrupted by one traffic light. - U-turn at the fly-over or at the end of the A270, to return the same way to the Automotive Campus.	Platoon formation and platooning, without live traffic light data included in planner, but with automatic speed advice. - No live traffic light data available for planner - Driver uses the Android app if Technolution gives a flexible route - Automatic speed advice also includes reaction to road-side camera-service objects (LDM closed loop) - Automatic speed advice is turned on, so does actuate the vehicle - Vehicle1 is starting at another location in Helmond (#2), vehicle2 at default location (Automotive Campus) - Platooning (CACC and lane keeping) on the A270 when possible.	371, 372, 373, 374, 376, 378, 379, 380, 381, 382, 384, 385, 386, 387
Platoon formation only. - Disabled live traffic light data included in planner - Not using the Android app - Starting at default locations	Platoon formation feasibility.	76, 77, 78, 79, 80, 82
Platoon formation with live traffic light data included in planner. - Enabled live traffic light data included in planner - Not using the Android app - Starting at default locations	Platoon formation improvement by traffic light data.	84, 85, 86, 87, 88, 89, 90, 103, 104, 108

Platoon formation and platooning, from Helmond to Eindhoven and back to the Automotive Campus. - Starting in urban area with speed limits of 15 and 30 km/h. - Driving East on the Europaweg with speed limits of 50 and 70 km/h. This includes 3 crossings with traffic lights. - Driving on the the N270, along the Automotive Campus. One crossing with traffic lights, just before the A270. - Driving on the A270 (speed limit 100 km/h). Interrupted by one traffic light. - U-turn at the fly-over or at the end of the A270, to return the same way to the Automotive Campus. Test Plan Link: \\tsn.tno.nl\data\sv\sv-085574\7. Workdocuments (WP-Kluis)\Kluis (WPs)\2018_Q4\Pilot plan Test Plan Identifier: Pilot test protocol - v5	Platoon formation and platooning, with live traffic light data included in planner. - Live traffic light data available for planner - Driver uses the Android app - Starting at default locations - Platooning (CACC and lane keeping) on the A270 when possible.	120, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147
Platoon formation with live traffic light data included in planner. - Enabled live traffic light data included in planner - Not using the Android app - Starting at default locations - This test was filmed, including the GUI.	Platoon formation improvement by traffic light data.	101
Driving without speed advice or any services to measure base-line. Comment: Baseline	Baseline	99
Platoon formation with live traffic light data included in planner. - Enabled live traffic light data included in planner - Not using the Android app - Starting at default locations - Used for pieces of the film (to be made)	Platoon formation variation testing.	97
Platoon formation, starting from different locations, with live traffic light data included in planner. - Enabled live traffic light data included in planner - Not using the Android app - Starting at different locations	Platoon formation variation testing.	96

Platoon formation, starting from different locations, with live traffic light data included in planner. - Enabled live traffic light data included in planner - Not using the Android app - Starting at different locations	Platoon formation variation testing.	91, 92, 93, 94, 95, 102, 105, 106, 107

10.2.2 Brainport, automated valet parking

Applicable datasets references
AUTOPILOT_BrainPort_AutomatedValetParking_DriverVehicleInteraction AUTOPILOT_BrainPort_AutomatedValetParking_DroneAvpCommand AUTOPILOT_BrainPort_AutomatedValetParking_EnvironmentSensorsAbsolute AUTOPILOT_BrainPort_AutomatedValetParking_EnvironmentSensorsRelative AUTOPILOT_BrainPort_AutomatedValetParking_IotVehicleMessage AUTOPILOT_BrainPort_AutomatedValetParking_ParkingSpotDetection AUTOPILOT_BrainPort_AutomatedValetParking_PositioningSystem AUTOPILOT_BrainPort_AutomatedValetParking_PositioningSystemResampled AUTOPILOT_BrainPort_AutomatedValetParking_Vehicle AUTOPILOT_BrainPort_AutomatedValetParking_VehicleAvpCommand AUTOPILOT_BrainPort_AutomatedValetParking_VehicleAvpStatus AUTOPILOT_BrainPort_AutomatedValetParking_VehicleDynamics

Scenario description	Session description	CTS ID list
The AD-vehicle receives parking command message (AutoPilot.VehicleCommand) containing the destination parking spot and the free obstacle route and drives from the drop-off position and parks to the destination parking spot. During the parking process the vehicle send two type of messages (AutoPilot.PositionEstimate and AutoPilot.VehicleAVPStatus)	The dropoff Scenario with TNO vehicle. TNO vehicle parks autonomously from the dropoff location to the selected parking spot at the parking area on the automotive campus.	322, 323, 324, 325, 326
RS Camera object detection and publication of the iot message from type AutoPilot.ObjectDetection to the PMS via IoT platforms	A car drive on the AVP road segment and stop and the RS camera detect the car as obstacle and send the obstacle information to the PMS for free obstacle route calculation for AD-car	117

The AD-vehicle receives parking command message (AutoPilot.VehicleCommand) containing the destination parking spot and the free obstacle route and drives from the drop-off position and parks to the destination parking spot. During the parking process the vehicle send two type of messages (AutoPilot.PositionEstimate and AutoPilot.VehicleAVPStatus)	The dropoff Scenario with DLR vehicle. DLR vehicle parks autonomously from the dropoff location to the selected parking spot at the parking area on DLR test area in Brunswick.	523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537
RS Camera parking spot occupancy detection and publication of the iot message from type AutoPilot.ParkingSpotDetection to the PMS via IoT platforms	A AD-car parks to the selected parking spot the rs camera detect the car at the parking spot and publish the occupancy information to the PMS for parking management purpose	116
The Vehicle received parking command message (AutoPilot.VehicleCommand) containing the destination parking spot and the free obstacle route and drives from the drop-off position and parks to the destination parking spot. During the parking process the vehicle send the two type of messages (AutoPilot.PositionEstimate and AutoPilot.VehicleAVPStatus)	The TNO vehicle parks autonomously from the dropoff to the selected parking spot in the parking place at the automotive campus.	112, 113, 114, 115
The Vehicle received parking command message (AutoPilot.VehicleCommand) containing the destination parking spot and the free obstacle route and drives from the drop-off position and parks to the destination parking spot. During the parking process the vehicle send the two type of messages (AutoPilot.PositionEstimate and AutoPilot.VehicleAVPStatus)	The TNO vehicle parks autonomously from the dropoff to the selected parking spot in the parking place at the automotive campus. the	109
The Vehicle received parking command message (AutoPilot.VehicleCommand) containing the destination parking spot and the free obstacle route and drives from the drop-off position and parks to the destination parking spot. During the parking process the vehicle send the two type of messages (AutoPilot.PositionEstimate and AutoPilot.VehicleAVPStatus)	The TNO vehicle parks autonomously from the dropoff to the selected parking spot in the parking place at the automotive campus.	110, 111

The drone receives AVP command message (Message type AutoPilot.DroneAVPCommand) from PMS via the IBM IoT Platfom. The command message contains the instruction about the selected parking spots to be checked. The drone takes off and fly to the corresponding parking spots detects the occupancy (â€œFREEâ€• or â€œOCCUPIEDâ€•) of the parking spot and publishes the message from type AutoPilot.ParkingSpotDetection to the PMS via IBM Watson IoT platform und return to the lading position and landed. During the flight the drone sends continuously the message about it current position and some status information as message from type AutoPilot.PositionEstimate to the PMS via IoT Platform.	Selection of one free parking spot to be check (see command message contain), the drone detects the parking spot and publish the occupancy information to the PMS for parking management purpose	118, 119
The AD-vehicle receives parking command message (AutoPilot.VehicleCommand) containing the destination pickup location and the free obstacle route and drives from the parking spot and parks to the destination pickup location. During the collection process the vehicle sends two type of messages (AutoPilot.PositionEstimate and AutoPilot.VehicleAVPStatus)	The pickup Scenario with TNO vehicle. TNO vehicle parks autonomously from the parking spot to the pickup location at the parking area on the automotive campus.	327, 328, 329, 330, 331, 332
The AD-vehicle receives parking command message (AutoPilot.VehicleCommand) containing the destination pickup spot and the free obstacle route and drives from the parking spot and parks to the destination pickup spot. During the collection process the vehicle send two type of messages (AutoPilot.PositionEstimate and AutoPilot.VehicleAVPStatus)	The pickup scenario with DLR vehicle. DLR vehicle parks autonomously from the parking spot to the destination pickup spot at the parking area on DLR test site in Brunswick.	538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556

10.2.3 Brainport, highway pilot

Applicable datasets references
AUTOPILOT_BrainPort_HighwayPilot_AdasCommand
AUTOPILOT_BrainPort_HighwayPilot_Anomaly
AUTOPILOT_BrainPort_HighwayPilot_AnomalyImage
AUTOPILOT_BrainPort_HighwayPilot_DriverVehicleInteraction
AUTOPILOT_BrainPort_HighwayPilot_EnvironmentSensorsAbsolute
AUTOPILOT_BrainPort_HighwayPilot_EnvironmentSensorsRelative
AUTOPILOT_BrainPort_HighwayPilot_Hazard
AUTOPILOT_BrainPort_HighwayPilot_IotVehicleMessage
AUTOPILOT_BrainPort_HighwayPilot_PositioningSystem
AUTOPILOT_BrainPort_HighwayPilot_Vehicle
AUTOPILOT_BrainPort_HighwayPilot_VehicleDynamics

Scenario description	Session description	CTS ID list
The road side camera detects objects (obstacle) on the test track and publishes the ANO IoT message.	10 minute observation of a section of the Automotive Campus Test Track	167, 170, 281, 287
The detection car drives around the track in manual mode, at an average speed close to laps for detections, but with all detections systems disabled.	25 laps with VW Tiguan on Automotive Campus Test Track	164, 277
The detection car drives around the track in manual mode, with Camera and IMU detection ON.	25 laps with VW Tiguan on Automotive Campus Test Track	166, 278, 280
The driving adaptation car drives around the track in simulated autonomous mode (ACC) and applies ADASINs at Hazards locations.	25 laps with Jaguar F-Pace on Automotive Campus Test Track	288, 289
The detection car drives around the track in manual mode, with Camera detection ON.	25 laps with VW Tiguan on Automotive Campus Test Track	284
The driving adaptation car drives around the track in simulated autonomous mode (ACC).	18 laps with Jaguar F-Pace on Automotive Campus Test Track	282, 169
The detection car drives around the track in manual mode, with IMU detection ON.	25 laps with VW Tiguan on Automotive Campus Test Track	165
The driving adaptation car is driven around the track in manual mode, but driving instructions are communicated to the driver.	12 laps with Jaguar F-Pace on Automotive Campus Test Track	171, 283

Hazards have been detected and populate the map. Control Center publishes ADAS Instructions for some hazards.	Control Center published ADAS Instructions for 4 identified hazards of the track.	285, 290

10.2.4 Brainport, urban driving

Applicable datasets references
AUTOPILOT_BrainPort_UrbanDriving_DriverVehicleInteraction
AUTOPILOT_BrainPort_UrbanDriving_EAI2Mobile
AUTOPILOT_BrainPort_UrbanDriving_EnvironmentSensorsAbsolute
AUTOPILOT_BrainPort_UrbanDriving_EnvironmentSensorsRelative
AUTOPILOT_BrainPort_UrbanDriving_IOT_CEMA_Message
AUTOPILOT_BrainPort_UrbanDriving_IOT_FlowRadar_Message
AUTOPILOT_BrainPort_UrbanDriving_IotVehicleMessage
AUTOPILOT_BrainPort_UrbanDriving_IOT_VehicleStatus
AUTOPILOT_BrainPort_UrbanDriving_PositioningSystem
AUTOPILOT_BrainPort_UrbanDriving_SmartphoneGPS
AUTOPILOT_BrainPort_UrbanDriving_SmartphoneStatus
AUTOPILOT_BrainPort_UrbanDriving_TaxiRequest
AUTOPILOT_BrainPort_UrbanDriving_Vehicle
AUTOPILOT_BrainPort_UrbanDriving_VehicleDynamics

Scenario description	Session description	CTS ID list
Smartphone 3120 sends request, route 2 is being blocked by crowd (depending on run this changes), vehicle receives correct route and drives from west to east. Underway 2 VRU (3121 & 3122) crosses the road.	Complete use case: Request vehicle, choose route on CEMA and GeoFenching VRU detection with 2 smartphone detection	579, 581, 591, 594, 596, 597, 598
Only GeoFenching VRU detection with 3 smartphone detection Test detection of multiple VRUs close to each other and compare with camera detections. Test different size GeoFence area (20m wide x 50m long) of detection with different pedestrian walking paths (for pedestrian prediction) see Test plan Table 4	Route is fixed, vehicle drives north - south. Underway 1 group of 3 VRU crosses the road.	635, 632, 633, 634, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657

Complete use case: Request vehicle, choose route on CEMA and GeoFenching VRU detection with 4 smartphone detections Test detection of multiple VRUs close to each other and compare with camera detections. Test GeoFence area of detection with different pedestrian walking paths (for pedestrian prediction) Make comparison with GeoFenching enabled (runs 1-12) and with GeoFencing disabled (runs 13â€“25)	Smartphone 3120 sends request, route 2 is being blocked by crowd (depending on run this changes), vehicle receives correct route and drives from west to east. Underway 2 groups of each 2 VRU crosses, or walk alongside the road (see Test Plan Table 3 for exact walking patterns).	608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630
Base line test: no CEMA, no GeoFencing enabled. Vehicle only brakes on camera detection, when vehicle is blocked on its route by a crowd	-	602, 603, 604, 605, 606, 607, 582, 583, 599, 600, 601

Precondition: - AD Vehicle (3177) is at starting position - VRU with Smartphone (3120) is at end position - VRU with Smartphone (3121) and FlowRadar (3175) is positioned halfway the vehicle track - CEMA device is on route 2 and detecting crowd Initiate Rebalancing phase: 1. Smartphone 3120: send taxi request (send end location to vehicle (lon / lat)) Initiate Routing phase: 2. CEMA devices detect crowds on route 1 or route 2 --> dynamically change routes base on CEMA 3. Motion planning: Vehicle uses CEMA detection and call taxi request as input for choosing one of the two routes Initiate Urban Driving phase (incl. VRU detection) 4. Vehicle starts driving 5. Underway VRU (3121) is detected using smartphone and FlowRadar ITS-G5 6. With GeoFenching, vehicle starts slowing down on smartphone 7. With camera detection vehicle brakes when VRU is inline of camera 8. After VRU passes, vehicle continues journey 9. Vehicle arrives at requested end point (at VRU 3120).	Rebalancing: - relocate the vehicle on request Urban Driving: - rerouting on crowd estimation (CEMA) - VRU detection with GeoFenching in vehicle (speed reduction) and warning on VRU smartphone	172
Smartphone 3120 sends request, route 2 is being blocked by crowd (depending on run this changes), vehicle receives correct route and drives from west to east. Underway 1 VRU crosses the road.	Complete use case: Request vehicle, choose route on CEMA and GeoFenching VRU detection with 1 smartphone detection	558, 559, 560, 561, 562, 563, 564, 565, 567, 568, 580, 584, 585, 557, 566, 577, 570, 571, 572, 573, 574, 575, 576, 569, 578, 586, 587, 588, 589, 590, 592, 593, 595

Only GeoFenching VRU detection with 3 smartphone detection Test detection of multiple VRUs close to each other and compare with camera detections. Test different size GeoFence area (20m wide x 50m long) of detection with different pedestrian walking paths (for pedestrian prediction)	Route is fixed, vehicle drives north - south. Underway 1 group of 3 VRU crosses the road.	631

10.2.5 Livorno, urban driving

Applicable datasets references
AUTOPILOT_Livorno_UrbanDriving_Vehicle_all AUTOPILOT_Livorno_UrbanDriving_V2X_all AUTOPILOT_Livorno_UrbanDriving_IoT_all

Scenario description	Session description	CTS ID list
Detect a pedestrian crossing with the red light and send DENM message to warn vehicles of the presence of the pedestrian	Baseline	210, 209
Test session for fallen bicycle with AD and connected vehicle	The fallen bicycle use case aims to demonstrate the possibility for a vehicle to detect in advance, using V2X communication, the presence of a fallen bicycle on the road. In case of fall, the bicycle signals its presence to the other vehicles using DENM messages. The AD car safely reduces its speed and stops.	305, 306, 307, 308, 309, 310, 316
Test session for connected car detecting potholes using the combination of one or more of the following sensors: smartphone, 6LoWPAN vibration sensor, IMU. The information is sent to the cloud and can be sent back to other connected vehicles for warning. The information is also transmitted via V2V to AD cars that can automatically adapt the speed.	Test session with only a connected car with 6LoWPAN based pothole detector, lap of 1,9 km on the harbour's public road. Goal is to record data for the technical evaluation.	453, 454

Test session for connected car detecting potholes using the combination of one or more of the following sensors: smartphone, 6LoWPAN vibration sensor, IMU. The information is sent to the cloud and can be sent back to other connected vehicles for warning. The information is also transmitted via V2V to AD cars that can automatically adapt the speed.	Test session with only a connected car with IMU based pothole detector, lap of 1,9 km on the harbour's public road. Goal is to record data for the technical evaluation.	455, 456
Test session for AD+connected car and connected cars approaching a fallen bicycle.	The fallen bicycle use case aims to demonstrate the possibility for a vehicle to detect in advance, using V2X communication, the presence of a fallen bicycle on the road. In case of fall, the bicycle signals its presence to the other vehicles using DENM messages. The AD car publishes the detected event to the oneM2M and safely reduces its speed until to stop. Goal is to record data for the technical evaluation.	513, 514, 515, 516, 517, 518, 519, 520, 521, 522
Test session for connected car detecting potholes using the combination of one or more of the following sensors: smartphone, 6LoWPAN vibration sensor, IMU. The information is sent to the cloud and can be sent back to other connected vehicles for warning. The information is also transmitted via V2V to AD cars that can automatically adapt the speed.	Test session with only a connected car with smartphone based pothole detector, lap of 1,9 km on the harbour's public road. Goal is to record data for the technical evaluation.	450, 451, 452

Test session for AD+connected car and connected cars approaching an intersection regulated by a "smart" traffic light with a stereocamera able to detect jaywalking.	A "smart" traffic light (with stereocamera) sends SPaT and MAP messages describing the topology and the actual status of the traffic light. If a jaywalking occurrence is detected (pedestrian crossing with the red light) a DENM message is sent to warn vehicles of the presence of the pedestrian to other connected vehicles. The the hazard warning is also sent to the oneM2M platform on the cloud. An AD vehicle consumes the information and autonomously adapts its speed in order to cross the intersection without violating the traffic light phases, or even stop to avoid collision with pedestrian. The influence of other vehicles moving in front is considered too. Goal is to record data for the technical evaluation.	508, 509, 510, 511, 512, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507
Test session for fallen bicycle with connected but not automated vehicle	The fallen bicycle use case aims to demonstrate the possibility for a vehicle to detect in advance, using V2X communication, the presence of a fallen bicycle on the road. In case of fall, the bicycle signals its presence to the other vehicles using DENM messages.	177

Test session for AD vehicle approaching an intersection with jaywalking at traffic light	A "smart" traffic light with a stereocamera sends SPaT and MAP messages describing the topology, actual status of the traffic light, presence of pedestrian, and jaywalking occurrence to other connected vehicles (via DENM) and to the oneM2M platform on the cloud. An AD vehicle consumes the information and autonomously adapts its speed in order to cross the intersection without violating the traffic light phases, or even stop to avoid collision with pedestrian. The influence of other vehicles moving in front is considered too.	317, 318, 320, 321
Test session for AD vehicle approaching an intersection regulated by traffic light	A "smart" traffic light sends SPaT and MAP messages describing the topology, actual status of the traffic light to other connected vehicles and to the oneM2M platform on the cloud. An AD vehicle consumes the information and autonomously adapts its speed in order to cross the intersection without violating the traffic light phases, considering also other vehicles moving in front.	311, 312, 313, 314, 315
Test session for AD+connected car and connected cars approaching an intersection regulated by a "smart" traffic light.	A "smart" traffic light sends SPaT and MAP messages describing the topology, actual status of the traffic light to other connected vehicles and to the oneM2M platform on the cloud. An AD vehicle consumes the information and autonomously adapts its speed in order to cross the intersection without violating the traffic light phases, considering also other vehicles moving in front. Goal is to record data for the technical evaluation.	457, 458, 459, 460, 461, 462, 483, 484, 485, 486, 487, 488, 489, 491, 492, 493, 494, 495, 496, 497

The pedestrian detection use case aims to demonstrate the possibility for a vehicle to be informed, using V2X communication, of the presence of a pedestrian crossing the road when the traffic light is green for the vehicle. In case of presence of a pedestrian, the RSU with detection capabilities signals the presence of the pedestrian to the vehicles using DENM messages.	Pedestrian detection	178

10.2.6 Livorno, highway pilot

Applicable datasets references
AUTOPILOT_Livorno_HighwayPilot_Vehicle_all AUTOPILOT_Livorno_HighwayPilot_V2X_all AUTOPILOT_Livorno_HighwayPilot_IoT_all

Scenario description	Session description	CTS ID list
Precondition: 1. AD cars with C-eHorizon and V2X OBU devices on board travels on the highway. The highway is equipped with IoT G5 RSUs. All the devices publish and share the information by the oneM2M platform in the cloud. Actions or events: 1 The Traffic Control Center publishes the presence of roadway works to the OneM2M platform. 2 The RSU (subscribed to the OneM2M platform) receives the information and it broadcasts to the vehicles the DENM message containing information about available lanes, speed limits, geometry, alternative routes etc. 3 At the same time the CONTI cloud is subscribed to the oneM2M platform; it receives and share with the FCA cloud the information of the road works, updating dynamically the maps of the Connected e-Horizon installed onboard the CRF AD car 4 The in-vehicle application fusing the information from the OBU, the C-eHorizon and on-board sensors, performs speed adaptation and lane change maneuvers Relevant situations: How the AD function interacts with different IoT input: from I2V (DENM, Roadwork position and extention); from V2V (CAM with info from other vehicles).	Test session with one AD+connected car and tone connected car, lap of 11,6 km on the highway. Goal is to record data for the technical evaluation.	472, 474, 473

Precondition: 1. AD cars with C-eHorizon and V2X OBU devices on board travels on the highway. The highway is equipped with IoT G5 RSUs. All the devices publish and share the information by the oneM2M platform in the cloud. Actions or events: 1 The Traffic Control Center publishes the presence of roadway works to the OneM2M platform. 2 The RSU (subscribed to the OneM2M platform) receives the information and it broadcasts to the vehicles the DENM message containing information about available lanes, speed limits, geometry, alternative routes etc. 3 At the same time the CONTI cloud is subscribed to the oneM2M platform; it receives and share with the FCA cloud the information of the road works, updating dynamically the maps of the Connected e-Horizon installed onboard the CRF AD car 4 The in-vehicle application fusing the information from the OBU, the C-eHorizon and on-board sensors, performs speed adaptation and lane change maneuvers Relevant situations: How the AD function interacts with different IoT input: from I2V (DENM, Roadwork position and extention); from V2V (CAM with info from other vehicles).	Test session with one AD+connected car and two connected cars, lap of 11,6 km on the highway. Goal is to record data for the technical evaluation.	481, 482
Precondition: 1. AD cars with C-eHorizon and V2X OBU devices on board travels on the highway. The highway is equipped with IoT G5 RSUs. All the devices publish and share the information by the oneM2M platform in the cloud. Actions or events: 1 The Traffic Control Center publishes the presence of roadway works to the OneM2M platform. 2 The RSU (subscribed to the OneM2M platform) receives the information and it broadcasts to the vehicles the DENM message containing information about available lanes, speed limits, geometry, alternative routes etc. 3 At the same time the CONTI cloud is subscribed to the oneM2M platform; it receives and share with the FCA cloud the information of the road works, updating dynamically the maps of the Connected e-Horizon installed onboard the CRF AD car 4 The in-vehicle application fusing the information from the OBU, the C-eHorizon and on-board sensors, performs speed adaptation and lane change maneuvers Relevant situations: How the AD function interacts with different IoT input: from I2V (DENM, Roadwork position and extention); from V2V (CAM with info from other vehicles).	Test session with one AD+connected car and tone connected car, lap of 11,6 km on the highway. Goal is to record data for the technical evaluation.	475

Precondition: 1. AD cars with C-eHorizon and V2X OBU devices on board travels on the highway. The highway is equipped with IoT G5 RSUs. All the devices publish and share the information by the oneM2M platform in the cloud. Actions or events: 1 The Traffic Control Center publishes the presence of roadway works to the OneM2M platform. 2 The RSU (subscribed to the OneM2M platform) receives the information and it broadcasts to the vehicles the DENM message containing information about available lanes, speed limits, geometry, alternative routes etc. 3 At the same time the CONTI cloud is subscribed to the oneM2M platform; it receives and share with the FCA cloud the information of the road works, updating dynamically the maps of the Connected e-Horizon installed onboard the CRF AD car 4 The in-vehicle application fusing the information from the OBU, the C-eHorizon and on-board sensors, performs speed adaptation and lane change maneuvers Relevant situations: How the AD function interacts with different IoT input: from I2V (DENM, Roadwork position and extention); from V2V (CAM with info from other vehicles).	Test session with one AD+connected car and two connected cars, lap of 11,6 km on the highway. Goal is to record data for the technical evaluation.	478, 476, 477
Precondition: 1. AD cars with C-eHorizon and V2X OBU devices on board travels on the highway. The highway is equipped with IoT G5 RSUs. All the devices publish and share the information by the oneM2M platform in the cloud. Actions or events: 1 The Traffic Control Center publishes the presence of roadway works to the OneM2M platform. 2 The RSU (subscribed to the OneM2M platform) receives the information and it broadcasts to the vehicles the DENM message containing information about available lanes, speed limits, geometry, alternative routes etc. 3 At the same time the CONTI cloud is subscribed to the oneM2M platform; it receives and share with the FCA cloud the information of the road works, updating dynamically the maps of the Connected e-Horizon installed onboard the CRF AD car 4 The in-vehicle application fusing the information from the OBU, the C-eHorizon and on-board sensors, performs speed adaptation and lane change maneuvers Relevant situations: How the AD function interacts with different IoT input: from I2V (DENM, Roadwork position and extention); from V2V (CAM with info from other vehicles).	Test session with one AD+connected car and two connected cars, lap of 11,6 km on the highway. Goal is to record data for the technical evaluation.	480, 479

Precondition: 1. AD cars with C-eHorizon and V2X OBU devices on board travels on the highway. The highway is equipped with IoT G5 RSUs. All the devices publish and share the information by the oneM2M platform in the cloud. Actions or events: 1 The Traffic Control Center publishes the presence of roadway works to the OneM2M platform. 2 The RSU (subscribed to the OneM2M platform) receives the information and it broadcasts to the vehicles the DENM message containing information about available lanes, speed limits, geometry, alternative routes etc. 3 At the same time the CONTI cloud is subscribed to the oneM2M platform; it receives and share with the FCA cloud the information of the road works, updating dynamically the maps of the Connected e-Horizon installed onboard the CRF AD car 4 The in-vehicle application fusing the information from the OBU, the C-eHorizon and on-board sensors, performs speed adaptation and lane change maneuvers Relevant situations: How the AD function interacts with different IoT input: from I2V (DENM, Roadwork position and extention); from V2V (CAM with info from other vehicles).	pre-test session with only connected cars, lap of 11,6 km on the highway. Goal is to check all the system and data managment before the last iteration in May.	293, 292, 295, 294
Precondition: 1. AD cars with C-eHorizon and V2X OBU devices on board travels on the highway. The highway is equipped with IoT G5 RSUs. All the devices publish and share the information by the oneM2M platform in the cloud. Actions or events: 1 The Traffic Control Center publishes the presence of roadway works to the OneM2M platform. 2 The RSU (subscribed to the OneM2M platform) receives the information and it broadcasts to the vehicles the DENM message containing information about available lanes, speed limits, geometry, alternative routes etc. 3 At the same time the CONTI cloud is subscribed to the oneM2M platform; it receives and share with the FCA cloud the information of the road works, updating dynamically the maps of the Connected e-Horizon installed onboard the CRF AD car 4 The in-vehicle application fusing the information from the OBU, the C-eHorizon and on-board sensors, performs speed adaptation and lane change maneuvers Relevant situations: How the AD function interacts with different IoT input: from I2V (DENM, Roadwork position and extension); from V2V (CAM with info from other vehicles).	pre-test session with only connected cars, lap of 11,6 km on the highway. Goal is to check all the system and data managment before the last iteration in May.	240, 291

Precondition: A vehicle is driving in the first lane of a “smart highway" at 90 km/h with all the devices working correctly and connected to all services needed. Actions or events: 1 The puddle monitoring system of the highway trigger an puddle hazard warning for a specific extended zone. 2 The AD car receives the information by IoT based services and sets a speed limitation according to the area interested by hazard conditions: it smoothly decelerates in order to enter in the area at the proper speed. 3 At the end of dangerous area, as notified by the «smart road», the vehicle will recover the legally allowed cruise speed. Relevant situations: How the AD function interacts with different IoT input: from oneM2M platform (advisory speed limit due to puddles); from I2V (DENM, puddle hazard warning); from V2V (CAM with info from other vehicles).	pre-test session with only connected cars, lap of 12,3 km on the highway. Goal is to check all the system and data managment before the last iteration in May.	245, 227, 228, 229, 230, 231, 244, 246
Precondition: A vehicle is driving in the first lane of a «smart highway» at 90 km/h with all the devices working correctly and connected to all services needed. Actions or events: 1 The puddle monitoring system of the highway trigger a puddle hazard warning for a specific extended zone. 2 The AD car receives the information by IoT based services and sets a speed limitation according to the area interested by hazard conditions: it smoothly decelerates in order to enter in the area at the proper speed. 3 At the end of dangerous area, as notified by the «smart road», the vehicle will recover the legally allowed cruise speed. Relevant situations: How the AD function interacts with different IoT input: from oneM2M platform (advisory speed limit due to puddles); from I2V (DENM, puddle hazard warning); from V2V (CAM with info from other vehicles).	pre-test session with only connected cars, lap of 12,3 km on the highway. Goal is to check all the system and data managment before the last iteration in May.	239, 238, 237
Precondition: A vehicle is driving in the first lane of a «smart highway» at 90 km/h with all the devices working correctly and connected to all services needed. Actions or events: 1 The puddle monitoring system of the highway trigger an puddle hazard warning for a specific extended zone. 2 The AD car receives the information by IoT based services and sets a speed limitation according to the area interested by hazard conditions: it smoothly decelerates in order to enter in the area at the proper speed. 3 At the end of dangerous area, as notified by the «smart road», the vehicle will recover the legally allowed cruise speed. Relevant situations: How the AD function interacts with different IoT input: from oneM2M platform (advisory speed limit due to puddles); from I2V (DENM, puddle hazard warning); from V2V (CAM with info from other vehicles).	pre-test session with only connected cars, lap of 12,3 km on the highway. Goal is to check all the system and data managment before the last iteration in May.	243, 242

Precondition: A vehicle is driving in the first lane of a «smart highway» at 90 km/h with all the devices working correctly and connected to all services needed. Actions or events: 1 The puddle monitoring system of the highway trigger an puddle hazard warning for a specific extended zone. 2 The AD car receives the information by IoT based services and sets a speed limitation according to the area interested by hazard conditions: it smoothly decelerates in order to enter in the area at the proper speed. 3 At the end of dangerous area, as notified by the «smart road», the vehicle will recover the legally allowed cruise speed.5 Relevant situations: How the AD function interacts with different IoT input: from oneM2M platform (advisory speed limit due to puddles); from I2V (DENM, puddle hazard warning); from V2V (CAM with info from other vehicles).	pre-test session with only connected cars, lap of 12,3 km on the highway. Goal is to check all the system and data managment before the last iteration in May.	248, 247
Precondition: A vehicle is driving in the first lane of a "smart highway" at 90 km/h with all the devices working correctly and connected to all services needed. Actions or events: 1 The puddle monitoring system of the highway trigger an puddle hazard warning for a specific extended zone. 2 The AD car receives the information by IoT based services and sets a speed limitation according to the area interested by hazard conditions: it smoothly decelerates in order to enter in the area at the proper speed. 3 At the end of dangerous area, as notified by the "smart road", the vehicle will recover the legally allowed cruise speed. Relevant situations: How the AD function interacts with different IoT input: from oneM2M platform (advisory speed limit due to puddles); from I2V (DENM, puddle hazard warning); from V2V (CAM with info from other vehicles).	Test session with one AD+connected car and two connected cars, lap of 12,3 km on the highway. Goal is to record data for the technical evaluation.	470, 471, 469
Precondition: A vehicle is driving in the first lane of a "smart highway" at 90 km/h with all the devices working correctly and connected to all services needed. Actions or events: 1 The puddle monitoring system of the highway trigger a puddle hazard warning for a specific extended zone. 2 The AD car receives the information by IoT based services and sets a speed limitation according to the area interested by hazard conditions: it smoothly decelerates in order to enter in the area at the proper speed. 3 At the end of dangerous area, as notified by the "smart road", the vehicle will recover the legally allowed cruise speed. Relevant situations: How the AD function interacts with different IoT input: from oneM2M platform (advisory speed limit due to puddles); from I2V (DENM, puddle hazard warning); from V2V (CAM with info from other vehicles).	Test session with only connected cars, lap of 12,3 km on the highway. Goal is to check all the system and data managment before the next test session with AD cars.	463

Precondition: A vehicle is driving in the first lane of a "smart highway" at 90 km/h with all the devices working correctly and connected to all services needed. Actions or events: 1 The puddle monitoring system of the highway trigger an puddle hazard warning for a specific extended zone. 2 The AD car receives the information by IoT based services and sets a speed limitation according to the area interested by hazard conditions: it smoothly decelerates in order to enter in the area at the proper speed. 3 At the end of dangerous area, as notified by the "smart road", the vehicle will recover the legally allowed cruise speed. Relevant situations: How the AD function interacts with different IoT input: from oneM2M platform (advisory speed limit due to puddles); from I2V (DENM, puddle hazard warning); from V2V (CAM with info from other vehicles).	Test session with a AD+connected car, lap of 12,3 km on the highway. Goal is to record data for the technical evaluation.	464, 465
Precondition: A vehicle is driving in the first lane of a "smart highway" at 90 km/h with all the devices working correctly and connected to all services needed. Actions or events: 1 The puddle monitoring system of the highway trigger an puddle hazard warning for a specific extended zone. 2 The AD car receives the information by IoT based services and sets a speed limitation according to the area interested by hazard conditions: it smoothly decelerates in order to enter in the area at the proper speed. 3 At the end of dangerous area, as notified by the "smart road", the vehicle will recover the legally allowed cruise speed. Relevant situations: How the AD function interacts with different IoT input: from oneM2M platform (advisory speed limit due to puddles); from I2V (DENM, puddle hazard warning); from V2V (CAM with info from other vehicles).	Test session with a AD+connected car, lap of 12,3 km on the highway. Goal is to record data for the technical evaluation.	468, 467, 466

10.2.7 Versailles, platooning

Applicable datasets references
AUTOPILOT_Versailles_Platooning_Vehicle AUTOPILOT_Versailles_Platooning_V2X AUTOPILOT_Versailles_Platooning_IoT

Scenario description	Session description	CTS ID list

Automatic fleet rebalancing from one car sharing station to the other. Operator is driving the first car, following the recommendations of Traffic Light Assist service when approaching intersections. Followers are driving automatically.	Platooning between the two car sharing stations with 3 vehicles. Traffic lights of intersections 422 and 447 are real traffic lights, others are simulated, therefore decisions close to these traffic lights are not relevant.	334, 335, 338, 348, 388, 389, 390, 391
Automatic fleet rebalancing from one car sharing station to the other. Operator is driving the first car, following the recommendations of Traffic Light Assist service when approaching intersections. Followers are driving automatically.	Platooning between the two car sharing stations with 3 vehicles. Only real traffic lights of intersections 422 and 447 are recorded.	333, 336, 337

10.2.8 Versailles, urban driving

Applicable datasets references
AUTOPILOT_Versailles_UrbanDriving_Vehicle AUTOPILOT_Versailles_UrbanDriving_V2X AUTOPILOT_Versailles_UrbanDriving_IoT AUTOPILOT_Versailles_UrbanDriving_CAM

Scenario description	Session description	CTS ID list
Car sharing and urban driving with cyclist and pedestrian detection	Classic session: Booking of a vehicle at the car sharing station, manual driving through the city until reaching the part where the AD mode can be switched on. Then there are two rounds: the first one without IoT (VRUs not connected) then with IoT (pedestrian with smartphone/watch and cyclist with connected bike). Then manual driving back to car sharing station.	261, 262, 265, 267, 269, 270, 271, 272, 296
Cyclist and pedestrian detection	Automated Driving part of the trip only. Two rounds: the first one without IoT (VRUs not connected) then with IoT (pedestrian with smartphone/watch and cyclist with connected bike).	297, 273, 274, 275, 276

10.2.9 Vigo, urban driving

Applicable datasets references
AUTOPILOT_Vigo_Urban_Driving_Vehicle_all AUTOPILOT_Vigo_Urban_Driving_V2X_all AUTOPILOT_Vigo_Urban_Driving_IoT_all

Scenario description	Session description	CTS ID list
GLOSA and VRU detection	Human factor tests: 1 lap manually driving by user 2 laps autonomous driving without GLOSA and VRU detection 3 laps with GLOSA activated 1 lap with VRU detection activated	392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407

10.2.10 Vigo, automated valet parking

Applicable datasets references
AUTOPILOT_Vigo_Automated_Valet_Parking_Vehicle_all AUTOPILOT_Vigo_Automated_Valet_Parking_V2X_all AUTOPILOT_Vigo_Automated_Valet_Parking_IoT_all

Scenario description	Session description	CTS ID list
The vehicle starts parked in the drop off area. A drop off request is made to the parking service through the mobile phone application. Once the parking manoeuvre is finished, a pick up request is made. The test ends when the vehicle has been parked in the pick-up area and the app informs the user.	Human factor tests carried out with users in the parking lot of Praza do Rei en Vigo. The user made the drop off and pick up requests using the app from outside the vehicle.	179, 180, 181, 182, 183, 184