Road-, Air- and Water-based Future Internet Experimentation

Project Acronym:	**RAWFIE**		
Contract Number:	**645220**		
Starting date:	**Jan 1st 2015**	**Ending date:**	**Dec 31st, 2018**

Deliverable Number and Title	**D7.5 Data Management Plan (b)**		
Confidentiality	PU	**Deliverable type**[1]	R
Deliverable File	**D7.5_Data_Management_Plan(b)_v8.pdf**	**Date**	08.02.2017
Approval Status[2]	**WP leader, 1st Reviewer, 2nd Reviewer**	**Version**	2.0
Contact Person	**Stathes Hadjefthymiades**	**Organization**	UoA
Phone	+30 210 727 51 48	**E-Mail**	shadj@di.uoa.gr

[1] Deliverable type: P(Prototype), R (Report), O (Other)

[2] Approval Status: WP leader, 1st Reviewer, 2nd Reviewer, Advisory Board

AUTHORS TABLE

Name	Company	E-Mail
Stathes Hadjefthymiades	UoA	shadj@di.uoa.gr
Kakia Panagidi	UoA	kakiap@di.uoa.gr
Kostas Kolomvatsos	UoA	kostasks@di.uoa.gr
Miltos Kyriakakos	UoA	paskalis@di.uoa.gr
Philippe Dallemagne	CSEM	pda@csem.ch
Marcel Heckel	Fraunhofer	Marcel.Heckel@ivi.fraunhofer.de
Lionel Blondé	HESSO	lionel.blonde@hesge.ch
Giovanni Tusa	IES	g.tusa@iessolutions.eu
Nikolaos Priggouris	HAI	PRIGGOURIS.Nikolaos@haicorp.com

REVIEWERS TABLE

Name	Company	E-Mail
Giovanni Tusa	IES	g.tusa@iessolutions.eu
Philippe Dallemagne	CSEM	Philippe.Dallemagne@csem.ch

DISTRIBUTION

Name / Role	Company	Level of confidentiality[3]	Type of deliverable
All		PU	R

[3] Deliverable Distribution: PU (Public, can be distributed to everyone), CO (Confidential, for use by consortium members only), RE (Restricted, available to a group specified by the Project Advisory Board).

CHANGE HISTORY

Version	Date	Reason for Change	Pages/Sections Affected
0.1	2017-01-02	ToC / Initial version	All
0.2	2017-01-14	Structure updated /assignments among partners	All Sections
0.3	2017-01-25	Finalisation of partners contributions	All Sections
0.4	2017-01-27	Structure of the document enhanced	All Sections
0.5	2016-01-30	Review started	All Sections
0.5	2016-02-03	Reviewed version, ready for refinements	All Sections
0.6	2016-02-08	The comments addressed by the involved partners. The deliverable was prepared for the submission.	All Sections
0.7	2016-02-09	Final version	

Abstract:

The deliverable provides the second version of the data management plan that RAWFIE project adopts. The plan follows the Horizon 2020 Work Program 2014-15 directives for Data Management Plan (DMP). The purpose of this deliverable is to provide updates to the data management life cycle of the project. The deliverable provides an outline of the data types the project generates, whether and how it exploits or makes them accessible for verification and re-use, and how it will curate and preserve them.

Keywords: data management plan, data collection, data types and standards, access policy and sharing

1

Table of Contents

Part II

List of Figures

List of Tables

Abbreviations

Abbreviation	Meaning
DMP	Data Management Plan
GML	Geography Markup Language
KML	KML - formerly Keyhole Markup Language
WMS	Web Map Service
WMTS	Web Map Tile Service
WFS	Web Feature Service
JSON	JavaScript Object Notation
GIS	Geographic Information System
PMML	Predictive Model Markup Language
EDL	Experiment Description Language
XML	Extensible Markup Language
UxV	Unmanned aerial/ground/surface Vehicle
API	Application Program Interface
EDL	Experiment Description Language
UGV	Unmanned Ground Vehicle
USV	Unmanned Surface Vehicle
UAV	Unmanned Aerial Vehicle
VGG	Visual Geometry Group
IRP	Intellectual Property Rights
OA	Open Access
APC	Article Processing Charges
EC	European Commission
OAIPMH	Open Archives Initiative Protocol for Metadata Harvesting
DOI	Digital Object Identifier
OMN	Open-Multinet
PMML	Predictive Model Markup Language
ESRI	Environmental Systems Research Institute
LDAP	Lightweight Directory Access Protocol

Part III: Main Section

1 Introduction

1.1 Scope of D7.5

The present document is the second in a series of three documents related to the RAWFIE Data Management policy. These documents define the rules applied to all datasets generated during the project. The purpose of “D7.5 (b) - Data Management plan” is to provide an overview of the main elements of the data management plan after the second year of the project. It also describes the policy adopted to grand access to the parties interested in the data generated by the RAWFIE platform during its development, tests and operation. Finally, it discusses the compliance of the RAWFIE data structure, management and policy with respect to the EU regulations and directives. This deliverable will be continuously updated throughout the lifespan of the project. Figure 1 presents the main steps and actions involved in a typical data management cycle as were described in the previous version of the deliverable.

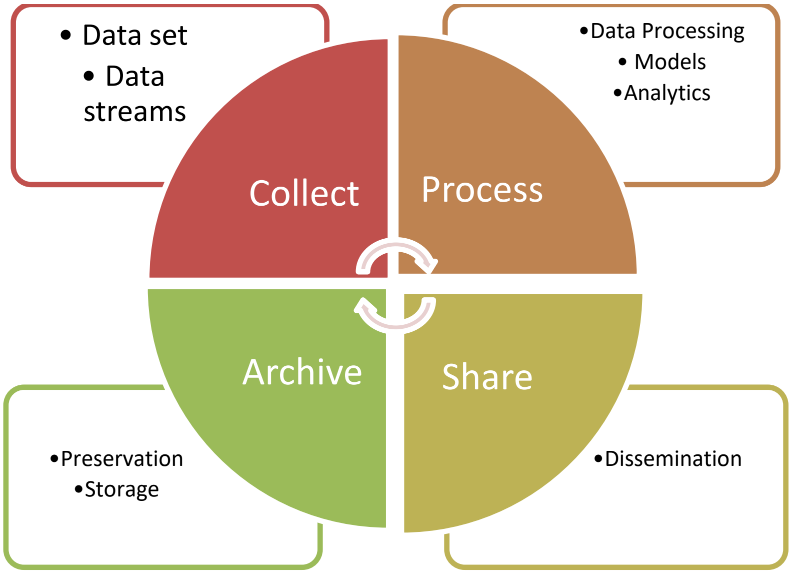

Figure 1: Data Management Cycle

The RAWFIE Data Management Plan (DMP) is realised in accordance with the Guidelines on Open Access to Scientific Publications and Research Data in Horizon 2020[4]. It also makes the first attempt to be compliant with the Guidelines on FAIR Data Management in Horizon 2020[5]. The compatibility with FAIR will be finalised in D7.6, i.e. the third version of the deliverable.

This document's structure is as follows: Section 2 gives an overview of the data description, data types and data processing in the RAWFIE ecosystem. Section 3 contains the data access procedure and the dissemination mechanisms that will take place to provide reusability and access in the future. In section 4 software tools for handling the data processing of research data during the execution of an experiment and the project lifetime are presented as well as standards and formats. Finally, Section 5 describes the procedures for the archiving and long-term storage.

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

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

2 Dataset reference and processing

RAWFIE data related to the execution of the experiments are distinguished in the following categories:

- ***Dynamic data***: this data refers to the data information that describes an UxV during an experiment in terms of system information, sensor types, central processing unit usage, storage usage, location etc.
- ***Static data***: this data refers to the characteristics of testbeds and resources. RAWFIE federation adds in advance static information like resource descriptions and properties, type of sensors on UxVs, UxV characteristics, testbed location, etc.
- ***Raw data***: data produced during the execution of the experiments. Any kind of sensor that participates in an experiment generates raw data. This data pushed to a message bus, which publishes them upon a request either from an experimenter or from a device that participates in the experiment and stores them in it for a short-time interval.
- ***Geospatial data***: this data refers to the geospatial information of data (data with a spatial reference or metadata) in the RAWFIE system. RAWFIE system will possibly generate and collect this data. Although this data is part of the static and dynamic data is presented separately (referenced in D7.4).

2.1.1 Dynamic Data from experiments

The RAWFIE UxV Protocol was devised to abstract the differences between UxVs and expose a simple, compact, extensible, and expressive interface to monitor and control UxVs in a platform agnostic way. The RAWFIE infrastructure can support the addition of new UxVs by creating adapters or translators to convert UxV specific information to the RAWFIE UxV Protocol. The reference frame of a UxV is defined in Table 1.

Table 1 - Dynamic Data Overview

Message	Data	Data types	Description
Header	sourceSystem sourceModule time	String string long	All messages of the UxV Message API contain the same header, used to encode basic information about the dispatching entity.
CPU Usage	header Value	Header Int	The amount of CPU resources that is currently in use.
Storage Usage	header available value	Header int int	Measurement of storage usage.
Fuel Usage	header value	Header Int	Amount of available fuel.

Location	header latitude longitude height n e d depth altitude	Header double double float double double double float, null float, null	The Location message encodes the position of the UxV in the World. It was designed to support all kinds of UxVs even when they are not capable of localizing themselves in the World. This message allows the UxV to encode its position in absolute (Latitude, Longitude, and Height) or relative (North/East/Down) coordinates. This message shall be published to the message bus and shall be consumed by any entity that needs to know the location of the UxV.
Attitude	header phi theta psi	Header float float float	Angles describing the attitude of a rigid body (i.e., Euler angles).
Linear Velocity	header x y z	Header float float float	Vector quantifying the direction and magnitude of the measured linear velocity that a system is exposed to.
Angular Velocity	header x y z	Header float float float	Vector quantifying the direction and magnitude of the measured angular velocity that a system is exposed to.
Linear Acceleration	header x y z	Header float float float	Vector quantifying the direction and magnitude of the measured linear acceleration that a system is exposed to.
Current	header value	Header float	Measurement of electrical current.
Voltage	header value	Header float	Measurement of electrical voltage.
Sensor Reading Scalar	header value Unit	Header float Unit	This message encodes scalar measurements of sensors.

Abort	Header	eu.rawfie.uxv.Headers	This command instructs the UxV to stop any executing actions and enter standby mode
Goto	header location speed timeout	eu.rawfie.uxv.Header eu.rawfie.uxv.Location float, null float	This command instructs a system to move to a given location at a given speed.
KeepStation	header location radius speed duration	eu.rawfie.uxv.Header eu.rawfie.uxv.Location float float, null float, null	This command instructs a system to keep station at a given location.

2.1.2 Static Data from experiments

Static data consists mainly of information related with the initial definition of an experiment. This information is usually defined prior to an experiment execution and may be updated after its completion. The term ‘static’ does not mean that the information is not updated, but mainly that it does not directly interfere with the actual data generated during the execution of an experiment. Static data mostly relate to the involved resources, sensor types, testbeds and scripts associated with the execution of an experiment as well as identifiers needed to identify or track an experiment within the RAWFIE platform. These static data are directly maintained/stored in (or can be extracted by) appropriate relational database tables defined at the platform level. Below in Table 2 there is a complete list of them appropriately categorized:

Table 2 - Static Data Overview

Data	Data type	Description
Experiment Related Data		
Experiment Id	String	Identifier for a defined experiment
Experiment Name	String	A (user friendly) name of the experiment
Experiment Description	String	A short description for the experiment
User Id	Integer	Internal identifier that can be used for obtaining additional information about the user that defined the experiment (i.e. name, surname etc.)
EDL script	String	Contains the EDL script initially defined for an experiment (information considered static since it is defined prior to the actual execution)
Testbed Id	String	Identifier of the testbed where the

		experiment is expected to take place (experiments cannot span multiple testbeds)
Resource Ids	String[]	Identifiers for the resources involved assigned in an experiment
Execution Related Data		
Execution Id	String	Identifier uniquely identifying an executing/executed experiment within the RAWFIE system
Start Execution	Timestamp	Timestamp denoting the start of execution
End Execution	Timestamp	Timestamp denoting the completion of execution
Experiment Status	Integer	Value indicates the execution status of an experiment (i.e. 0=BOOKED, 1= ONGOING, 2=COMPLETED). This field may be updated during the course of experiment execution
Reservation Related Data		
Reservation Id	String	Identifier of the user level reservation associated with an experiment
User Id	Integer	Internal identifier that can be used for obtaining additional information about the user that defined the reservation (i.e. name, surname etc.) This value should be the same with the <User Id> mentioned at the **Experiment Related Data** category
Resource Related Data[6]		
Resource Name	String	User friendly name of the resource
Resource Description	String	A short description for the resource
Resource Status	Integer	The latest status of the resource
Resource Type	Integer	Identifier denoting the type of the resource (i.e. UAV, UGV, USV etc.)

2.1.3 Raw data

The types of raw data generated by UxVs relate to the different sensor types that take part in the context of the RAWFIE project. We can classify the sensors and the relevant data into the following categories:

[6] These kind of data are not strictly related to a single experiment or its execution but we include them here for completeness

- Environmental sensors (temperature, thermal, heat, moisture, humidity, air pressure)
- Position, angle, displacement, distance, speed, acceleration
- Proximity (able to detect the presence of nearby objects)
- Navigation instruments

2.1.4 Geospatial data

Geospatial data appears in various formats and relations in the RAWFIE system. Sometimes the data itself has a spatial aspect, sometimes it is just metadata (i.e. descriptive data belonging to the original data). The following list (Table 3) gives an overview of the types of data with a spatial reference that will be possibly generated and / or collected inside RAWFIE.

Table 3 - Geospatial Data Overview

Data	Data type	Description
UxV location	Point	The location of an UxV during an experiment. Used in the Visualisation Engine
UxV course	Line	The current course an UxV is taking, i.e. an extrapolation of the current position together and its direction to know where the UxV will probably be in the next seconds or minutes.
Waypoints	Point[]	A time ordered list of waypoints for UxV navigation / predefined routes. They can have absolute coordinates or relative ones in respect to the current position (e.g. ‘move 30 meters in the direction of 45°’). Used for experiment authoring and in the resource controller during execution.
Geo-fence	Polygon	Regions where an event or alarm should be triggered when an UxV enters or leaves. Used in experiment authoring (EDL)
Sensor measurement location	Point	Location where a sensor measurement has been recorded. It is metadata for sensor data types.
Detected object	any	An object detected by sensors or evaluation of sensor values. The type of object highly depends on task which should be performed by UxV, e.g.:

		- border surveillance: intruders / potential threats - firefighting: trees, fire or empty space which would form a natural block to the spreading fire - monitoring of water canals: cracks in canal's wall structure The position or geo-referenced outline of the object is geospatial meta data of the experiment results
Testbed position or area	Point Polygon	The fixed location of the testbed (meta data). In the simple case it is just a coordinate, in the more precise case it is the area of the testbed. Used in experiment authoring (EDL) and resource exploring
Testbed surroundings	Any	The surroundings of a testbed. These could influence the experiments. Potential objects could be: - barriers (buildings, trees etc.) - streets - water ways - water surface - digital elevation model (above and under water) Used in experiment authoring (EDL) and resource exploring as well as for validation of experiments in their aftermath.

2.1.5 Processed data, models and analytics

Processed data refer to the outcome of models and statistical methods that will be generated by the stream analytics platform. Typical models include classification and outlier detection. Since most of our algorithms will be working on streaming data there is no specific model that can be used to generalize to all possible time instances. Instead we will be open-sourcing the architecture as is commonly done in the Deep Learning community. An example of this is the GoogleNet[7] & Visual Geometry Group[8] (VGG) style deep architectures. Providing the model architecture via a version control system such as Git[9] will allow for iterative development and reproduction by anyone who chooses to utilize these algorithms.

[7] http://image-net.org/challenges/LSVRC/2014/slides/GoogLeNet.pptx
[8] http://www.robots.ox.ac.uk/~vgg/
[9] https://git-scm.com/

The data mining and machine learning communities traditionally rely on the exchange and publication of datasets. This is achieved by a number of relevant data repositories which will be described in the next section, and much less through models. The publicly available datasets are used to compare and test different learning algorithms and it is one of the means that the community has used to ensure the replicability of the scientific results and the fair comparison of different learning methods.

Once the raw data for specific learning tasks are available, different teams can test their own algorithms on them. Probably the most well-known repository for datasets used by data mining and machine learning teams is the UCI machine learning repository [6]. We will discuss in more detail the availability and use of existing repositories in a following section on the dissemination of the data that will be generated by the project. Within the UCI repository one may find a number of datasets similar in nature to the data that will be generated within RAWFIE. These are mainly time-series datasets from different application domains such as finance, social media, physical activity sensors, chemical sensors and more. Nevertheless these datasets are not directly relevant for the RAWFIE project. Some of them might be used to provide additional testing datasets for the learning and mining algorithms that will be developed in RAWFIE.

3 Open Access of RAWFIE outcomes

The scientific and technical results of the RAWFIE project are expected to be of high interest for the scientific community. Throughout the duration of the project, RAWFIE partners may disseminate (subject to their legitimate interests) the obtained results and knowledge to the relevant scientific communities through contributions in journals and international conferences mainly in the field of IoT, wireless communications, robotics, etc. This dissemination of the research outcomes should be firstly secured with any relevant protection (e.g., Intellectual Property Rights (IPR)).

The RAWFIE project will also produce, transform and use data that is of interest and has a value for the next phases of the RAWFIE deployment on one hand and for other initiatives and contexts on the other hand.

This chapter addresses the access to these outcomes. Any of its publications will come after the more general decision on whether to go for a publication directly or to seek first protection by registering . If the Steering Committee decides that the scientific research will not be protected through IPR, but will rather be published directly, then the project is aware that Open Access must be granted to all scientific publications resulting from Horizon 2020 actions.

This will be done in accordance with the Guidelines on Open Access to Scientific Publications and Research Data in Horizon 2020. The process shown in Figure 2, was taken from the aforementioned document.

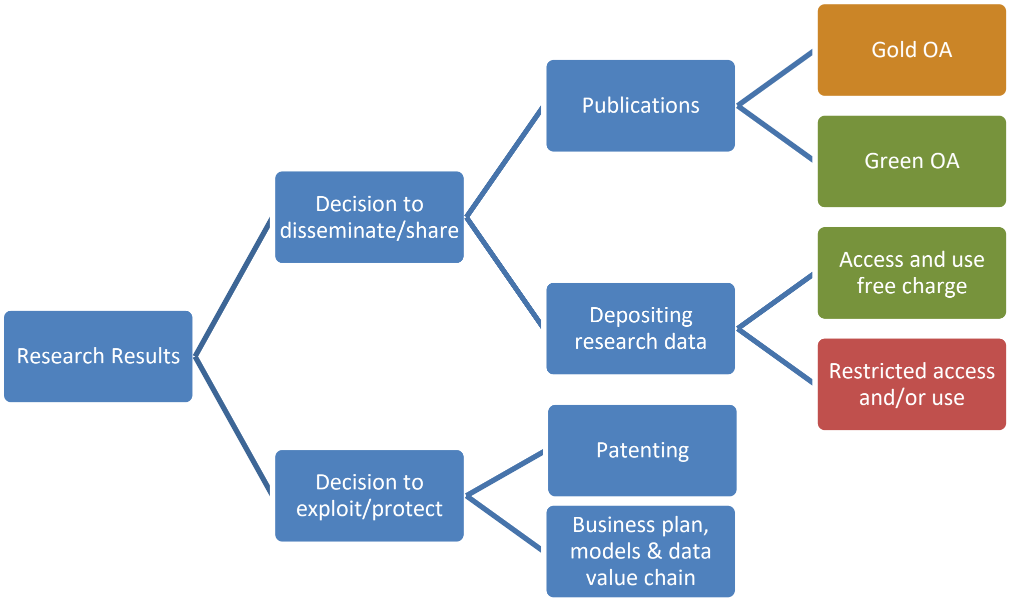
Research Results
Decision to disseminate/share
Decision to exploit/protect
Publications
Depositing research data
Patenting
Business plan, models & data value chain
Gold OA
Green OA
Access and use free charge
Restricted access and/or use

Figure 2: Process for handling access to research results

In the 'gold' Open Access (OA) approach of a peer-reviewed scientific research article, the scientific publisher immediately provides this article in Open Access mode. The associated costs shift away from readers. The most common business model is the one-off payment by authors. These costs, often referred to as Article Processing Charges (APCs), are usually paid by the researcher's university or research institute or the agency funding publishing the research. In other cases, subsidies or other funding models cover the costs of Open Access.

The 'green' Open Access approach to peer-reviewed scientific research articles means that the author, or a representative, self-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 to apply the Open Access mode only after an embargo period has elapsed. This embargo period is to allow the scientific publisher to recoup its investment by selling subscriptions and charging pay-per download/view fees during an exclusivity period.

3.1 Categories of RAWFIE data outputs for the Open Access mode

The following categories of RAWFIE outputs apply to a free of charge Open Access:

- Public Deliverables
- Conference/Workshop presentations (which may, or may not, be accompanied by papers, see below)
- Conference/Workshop papers and articles for specialist magazines; and
- Research (Experiment) Data and metadata

Furthermore, the provision of specific data sets to selected organisations will be possible in order to fulfil the H2020 requirements of "Grand Challenges"[10] for third parties to access, mine, exploit, reproduce and disseminate the results of the RAWFIE project.

The beneficiaries will have access to the information about the tools and instruments, for the sake of validating the results they will produce.

3.1.1 Open Access to RAWFIE Public Deliverables

3.1.1.1 Data Sharing

Open Access to the public deliverables will be achieved in RAWFIE by depositing the data into online repositories. The public deliverables will be stored in one or more of the following locations:

- The RAWFIE Web site[11], after approval by the Project Officer (if the document is subsequently updated, the original version will be replaced by the latest version)

[10] Known as "Societal Challenges": https://ec.europa.eu/programmes/horizon2020/en/h2020-section/societal-challenges

- The RAWFIE page on Cordis[12] web site, will host all public deliverables as submitted to the European Commission (EC).

3.1.1.2 Archiving and Preservation

Open Access to the project public deliverables will be maintained for at least 3 years following the project completion, through the Website.

3.1.1.3 Archived deliverables

The following table (Table 4) summarizes archived deliverables at 31/12/2016, which are available at the RAWFIE web page[7].

Table 4 - Archived deliverable

D3.1 - Specification & Analysis of RAWFIE Components Requirements (a)	WP3
D3.2 - Specification & Analysis of RAWFIE Components Requirements (b)	WP3
D4.1 - High Level Design and Specification of RAWFIE Architecture	WP4
D4.2 (a) - Design and Specification of RAWFIE Components	WP4
D4.4 - High Level Design and Specification of RAWFIE Architecture (2nd version)	WP4
D4.5 - Design and Specification of RAWFIE Components	WP4
D6.1: RAWFIE Operational Platform Testing and Integration Report (a)	WP6
D6.2: RAWFIE Platform Validation (a)	WP6
D7.1. Building the RAWFIE Community	WP7
D7.4 Data Management Plan (a)	WP7
D8.1: Open Calls, Report on Selection	WP8

[11] http://www.rawfie.eu/deliverables

[12] http://cordis.europa.eu/project/rcn/194297_en.html

3.1.2 Open Access to RAWFIE Conferences, Workshops and Presentations

3.1.2.1 Data Sharing

Open Access to conference/workshop presentations will be achieved in RAWFIE by depositing the data into an online research data repository. The presentations will be stored in the section of the promotion material in the RAWFIE Web site[13]

3.1.2.2 Archiving and Preservation

Open Access to project public presentations will be maintained for at least 3 years following the project completion, through the Website.

3.1.2.3 Archived Presentations

As of 31/12/2016, no presentation is released online.

3.1.3 Open Access to RAWFIE Publications

3.1.3.1 Data Sharing

As previously mentioned and described in section 1.1, there are two main routes to providing Open Access to these publications; namely; ´gold´ or´ green´. In any case, Open Access to its publications will be achieved in RAWFIE by depositing the data into online research data repositories. The publications will be stored in one or more of the following locations:

- An institutional research data repository
- The ZENODO[14] repository, operated by the EC through the funded OpenAIRE[15] project
- The RAWFIE Website[16]

The ZENODO repository is recommended by the EC's OpenAIRE initiative in order to unite all the research results arising from EC funded projects. ZENODO is an easy-to-use and innovative service that enables researchers, EU projects and research institutions to share and show case multidisciplinary research results (data and publications) that are not part of existing institutional or subject-based repositories. Namely, ZENODO enables users to:

- Easily share the long tail of small data sets in a wide variety of formats, including text, spreadsheets, audio, video, and images across all fields of science.
- Display and curate research results, got credited by making the research results citable, and integrate the min to existing reporting lines to funding agencies like the EC.
- Easily access and reuse shared research results.

[13] http://www.rawfie.eu/promotional-material
[14] https://zenodo.org/
[15] https://www.openaire.eu/
[16] http://www.rawfie.eu/publications

- Define the different licenses and access levels that will be provided.

Furthermore, ZENODO assigns a Digital Object Identifier[17] (DOI) to all publicly available uploads, in order to make the content easily and uniquely citable. This repository also makes use of the OAIPMH protocol (Open Archives Initiative Protocol for Metadata Harvesting) to facilitate the content search through the use of defined metadata. This metadata follows the schema defined in INVENIO3[18] (a free software suite enabling to run an own digital library or document repository on the web) and is exported in several standard formats such as MARCXML[19], Dublin Core[20] and Data Cite[21] Metadata Schema according to OpenAIRE Guidelines.

In addition, considering ZENODO as the repository, the short- and long-term storage of the research data will be secured since they are stored safely in same cloud infrastructure like research data from CERN's Large Hadron Collider[22]. Furthermore, it uses digital preservation strategies to storage multiple online replicas and to backup the files (Data files and metadata are backed upon a nightly basis).

Therefore, this repository fulfils the main requirements imposed by the EC for data sharing, archiving and preservation of the data generated in H2020 projects.

3.1.3.2 Publication Reference Identity (Digital Object Identifier-DOI)

The DOI uniquely identifies a document. The publisher, in the case that the document is included in the ´gold´ Open Access, or OpenAIRE, in the case that the document is archived in ZENODO, will allocate this identifier.

3.1.3.3 Archiving and Preservation

Open Access to project public presentations will be maintained for at least 3 years following the project completion, through the above repositories.

3.1.3.4 Archived Publications

1. K. Kolomvatsos, C. Anagnostopoulos, S. Hadjiefthymiades, ‘Distributed Localized Contextual Event Reasoning under Uncertainty’, accepted for publication in IEEE Internet of Things Journal, 2017, DOI 10.1109/JIOT.2016.2638119
2. Md Fasiul Alam, Stathes Hadjiefthymiades, Advanced, Hardware Supported In-Network Processing for the Internet of Things, to be presented in ICC 2017 (2nd international conference on Internet of things, Data and cloud computing), March 2017, Cambridge UK.

[17] http://www.doi.org/
[18] http://invenio.readthedocs.io/en/latest/
[19] http://www.loc.gov/standards/marcxml/
[20] http://dublincore.org/
[21] https://mds.datacite.org/
[22] https://home.cern/topics/large-hadron-collider

3.1.4 Open Access to RAWFIE Research Data

Apart from the Open Access to public deliverables, presentations and scientific publications, the Open Research Data Pilot also applies to two types of data:

- The data, including associated metadata, needed to validate the results presented in scientific publications (underlying data);
- Statistical data and metadata generated
    - in the course of the project, or
    - during the execution of experiments.

A lot of information generated during the experiments will form statistical data that will be used for the purpose of the dynamic tuning of the resources and plans. This data could also be used after the execution of the experiment (post-mortem), for diagnostics or further analysis of the experiment execution. This is the case for example in the network behaviour reporting, done through the analysis of the link quality, latency, throughput, etc. As experimental data contain information about the positions of UxVs, their operational measurements (cpu usage, battery consumption etc) and the sensor collected measurements RAWFIE consortium should take consideration about testbeds that are characterized as “sensitive areas” like Skaramagas naval base Staging processing is required as discussed further in section 5.2. After cleaning and filtering, they may also be used as reference data as described in Section 2.1.5. A complete description of the statistical information cannot be given here, but the main categories in which such data are generated are: *networking*, *processor* and *machine load*, *database transaction rates*, etc. In other words, beneficiaries will be able to choose data, additionally to the data underlying publications, they make available in Open Access mode.
According to this requirement, the underlying data related to the scientific publications will be made publicly available (see section3.1.1). This will allow that other researchers can make use of that information to validate the results, thus being a starting point for their investigations, as expected by the EC through its Open Access policy.

By design, RAWFIE will avoid any unnecessary collection of personal data. In cases where some limited personal data collection is required, each entity that accesses data commits itself to respect data confidentiality throughout its entire processing cycle. More explicitly, data should:

- Be fairly and lawfully processed;
- Be used for limited purposes;
- Be handled in an adequate, relevant and not excessive way;
- Be limited to what is needed and relevant for the research;
- Be collected on a voluntary basis under explicit consent from the end-users;
- Be accessed in aggregate form or anonymously;
- Be not kept longer than necessary;
- Be used in accordance with the data subject’s rights;
- Be processed without transferring it to countries with absent or insufficient data protection policies.

More generally, the RAWFIE platform is governed by the following principles, which should be respected by all users and partners:

- Respect of privacy, personal data protection and individual freedom of choice.
- Proportionality
  - By default, most RAWFIE experiments will avoid or limit the collection of personal data beyond what is necessary and relevant to the carried out for an experiment.
- Dissociation
  - If personal data is collected, the system will dissociate any identifying information, such as the email address, from the collected data.
- Principle of prior informed consent from the data originator.
- Protection of minors by restricting personal data collection from non-adults.
- Collected data are stored on Servers located in European countries.
- Collective responsibility
  - All users and stakeholders are required to respect data handling rules and to inform the project privacy officer of any detected attempt of privacy breach.
- Universality
  - The privacy and personal data protection standards followed by the RAWFIE platform are binding for the users and other interacting parties, regardless of their country of residence.
- No personal data is shared and transmitted to third parties, including governments and public agencies (except in cases of an, unlikely, judiciary decision).

4 Research Data – Tools and Standards

4.1 Tools

In this section, two major tools of the RAWFIE system are presented: The Apache Avro[23] tool which is a data serialization tool and provides a common framework for every robot to adhere to RAWFIE agnostically of their system through the adaptor of the UxV Node and the SAMANT ontology, an extension of Open-Multinet (OMN) ontology suit, which describes semantically the dynamic and static data of the RAWFIE ecosystem.

4.1.1 Apache AVRO formatted messages and Kafka Schema Registry

4.1.1.1 AVRO

According to its own documentation, the Apache Avro tool is a data serialization system with some useful capabilities. Avro provides:

- Rich data structures.
- A compact, fast, binary data format.
- A container file, to store persistent data.
- Remote procedure calls (RPC).
- Simple integration with dynamic languages. Code generation is not required to read or write data files nor to use or implement RPC protocols. Code generation as an optional optimization, only worth implementing for statically typed languages.

In order to use the Avro schemas, RAWFIE adopts the Apache Kafka based Confluent Platform[24] which provides an easy way to build real-time data pipelines and streaming applications. Having a single, central streaming platform for the RAWFIE infrastructure simplifies connecting data sources to Kafka, building applications with Kafka, as well as securing, monitoring, and managing your Kafka infrastructure.

Avro, being a schema-based serialization utility, accepts schemas as input. In spite of various schemas being available, Avro follows its own standards of defining schemas. These schemas describe the following details:

- type of file (record by default)
- location of record
- name of the record
- fields in the record with their corresponding data types

[23] https://avro.apache.org/docs/current/
[24] https://confluent.io/

Using these schemas, you can store serialized values in binary format using less space. These values are stored without the use of any metadata. Avro schemas are defined with JavaScript Object Notation (JSON)[25] document format, which is a lightweight text-based data interchange format. This facilitates implementation in languages that already have JSON libraries.

4.1.1.2 Kafka Schema Registry

One of the most important things is to manage the Avro schemas and how those schemas should evolve. A Kafka Schema Registry is adopted for that purpose which provides a serving layer for our metadata. It provides a RESTful interface for storing and retrieving Avro schemas. It stores a versioned history of all schemas, provides multiple compatibility settings and allows evolution of schemas according to the configured compatibility setting. It provides serializers that plug into Kafka clients and handle schema storage and retrieval for Kafka messages sent in the Avro format. Briefly, the Schema Registry:

- provides a serving layer for metadata.
- provides interface for storing and retrieving Avro schemas.
- stores a versioned history of all schemas, provides multiple compatibility settings and allows evolution of schemas according to the configured compatibility setting.
- provides serializers that plug into Kafka clients that handle schema storage and retrieval for Kafka messages that are sent in the Avro format.

In the end this Schema Registry is heavily based on the Java API of Confluent Schema Registry[26].

4.2 Ontologies for RAWFIE – 1st Open Call Winner

Over the past decade semantic information models have been regularly used to address interoperability issues on managing federated experimental infrastructures (e.g., NDL-OWL[27], NOVIIM[28], NML[29], INDL[30], etc.). One of the most recent efforts, the OWL encoded OMN ontology suite builds upon existing ontologies. OMN is still evolving supported by a community of experts within the FIRE and GENI community.

The ontology describes federated infrastructures and resources as generally as possible, while still supporting the management of their lifecycle in federated environments. OMN consists of a hierarchy of ontologies as depicted in Figure 3. The detailed description of the OMN ontology suite is provided in [17]. The OMN ontology at the highest level defines basic concepts and properties, which are then re-used and specialized in the subjacent ontologies. Included at every level are (i) axioms, such as the disjointness of each class; (ii) links to concepts in existing

[25] http://json.org/

[26] http://docs.confluent.io/2.0.0/schema-registry/docs/index.html

[27] https://staff.fnwi.uva.nl/j.j.vanderham/research/presentations/1111-Xin-SC11-semantic-bof.pdf

[28] http://www.fp7-novi.eu/

[29] https://tnc2013.terena.org/getfile/148

[30] http://staff.science.uva.nl/~vdham/research/publications/1212-INDL-report.pdf

ontologies, such as NML, INDL and NOVI; and (iii) properties that have been shown to be needed in related ontologies. In a nutshell:

- The Federation ontology describes federations, along with their members and related infrastructures.
- The Lifecycle ontology describes the whole lifecycle of resource/service management in the federation. This includes requests, reservation (schedule for allocation), provisioning and release.
- A resource in the OMN ontology is defined as any provisionable, controllable, and/or measurable entity. The Resource ontology augments the definitions of the Resource class defined in the main OMN upper ontology with concepts such as Node, Interface, Link, etc.
- The Component ontology covers concepts that are considered descendants of the Component class defined in the OMN upper ontology (e.g. CPU, Sensor, Core, Port, Image, etc.)
- A service is defined in the OMN ontology as any entity that has an API to use it. A service may further depend on a Resource. The Service ontology covers different services in the relevant application areas (e.g., Portal, etc.).
- The Monitoring ontology is directly linked to other OMN ontologies and facilitates interoperability in terms of enabling common monitoring data to be exchanged federation wide. It is built based on existing ontologies, such as the NOVI monitoring ontology. The OMN ontology suite is designed in a flexible, extensible way to cover specific domains.

Examples of such domains include wireless (e.g., Wi-Fi or sensors), SDN, Cloud computing, etc.

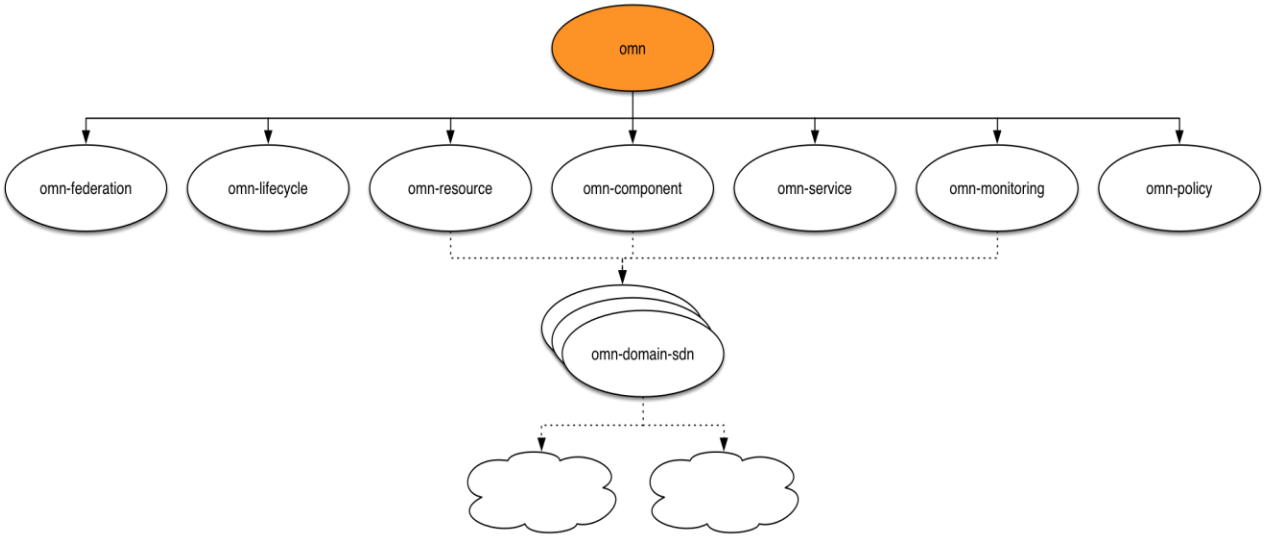

Figure 3: Open - Multinet ontology suite

4.2.1 SAMANT OMN Extended Ontology

The extension of the OMN ontology for the description of the resources of RAWFIE is twofold. It adopts many concepts from the ontologies of the OMN suite and includes two new ontologies to cover specifically the domains of UxVs and sensors. Furthermore, these ontologies include concepts from other existing relevant ontologies on sensors and measurements.

4.2.1.1 OMN UxV Ontology

Figure 4 illustrates the structure of OMN UxV (omn-domain-uxv) ontology. This ontology is available in Turtle[31] format.

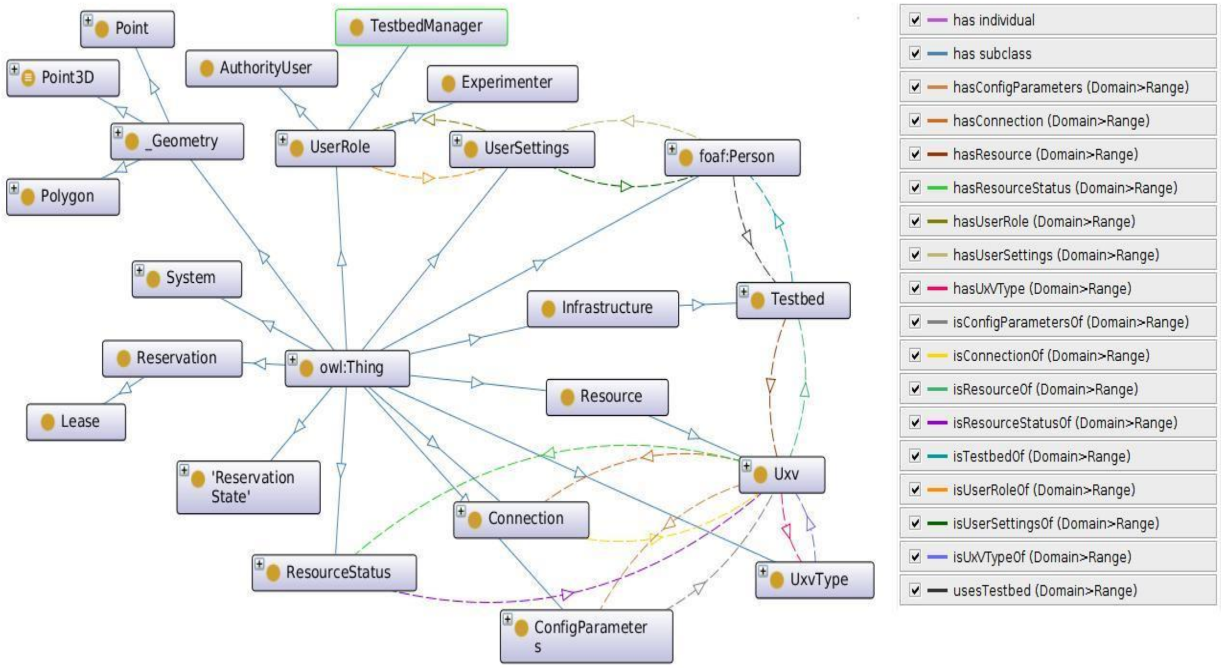
Point
TestbedManager
Point3D
AuthorityUser
Experimenter
_Geometry
UserRole
UserSettings
foaf:Person
Polygon
System
Infrastructure
Testbed
Reservation
owl:Thing
Resource
Lease
'Reservation State'
Uxv
Connection
ResourceStatus
UxvType
ConfigParameters
has individual
has subclass
hasConfigParameters (Domain>Range)
hasConnection (Domain>Range)
hasResource (Domain>Range)
hasResourceStatus (Domain>Range)
hasUserRole (Domain>Range)
hasUserSettings (Domain>Range)
hasUxVType (Domain>Range)
isConfigParametersOf (Domain>Range)
isConnectionOf (Domain>Range)
isResourceOf (Domain>Range)
isResourceStatusOf (Domain>Range)
isTestbedOf (Domain>Range)
isUserRoleOf (Domain>Range)
isUserSettingsOf (Domain>Range)
isUxVTypeOf (Domain>Range)
usesTestbed (Domain>Range)

Figure 4: OMN UxV ontology

This ontology describes the resources of RAWFIE testbeds, their reservation lifecycle and the attributes of RAWFIE members. Each RAWFIE testbed is described by the Testbed class that includes all the attributes of RAWFIE testbeds (name, description, location and UxV support) is linked with User Class and UxV class. The User class describes RAWFIE members and includes personal information and their role on RAWFIE testbeds. The UxV class describes the resources of each RAWFIE testbed. More specifically, it contains basic information about UxVs (name, description, location, UxV type) and is linked with many classes that describe the features of UxVs. The Connection class represents the communication capabilities of each UxV. The Resource Status class describes the current availability status of UxVs. The Health Status class describes the health status of UxVs and the Config Parameters class includes specific configuration parameters of each UxV. The reservation status of each UxV is described by the

[31] http://www.w3.org/TR/turtle/

Lease class. UxV class is linked with the System class of OMN Sensor Ontology, which describe the specification of the sensors attached on UxVs.

Figure 5 depicts the description of a ground unmanned vehicle (UgV) named UgV1. Ugv1 is part of the UgV Testbed (testbed) and is a type of UgV. Its connection features and configuration parameters are described by the UgV Connection and UgV Config Parameters individuals respectively. The health status of UgV1 is defined by term “OK” and UgV1 Health Information individuals. Its resource status is described by “Sleep Mode” status. The UgV1 Lease individual includes its reservation status. UgV1 Point 3D describes the exact location of UgV1 in terms of latitude, longitude and altitude. Finally Ugv1 Sensor System contains all the information of the attached sensors of UgV1.

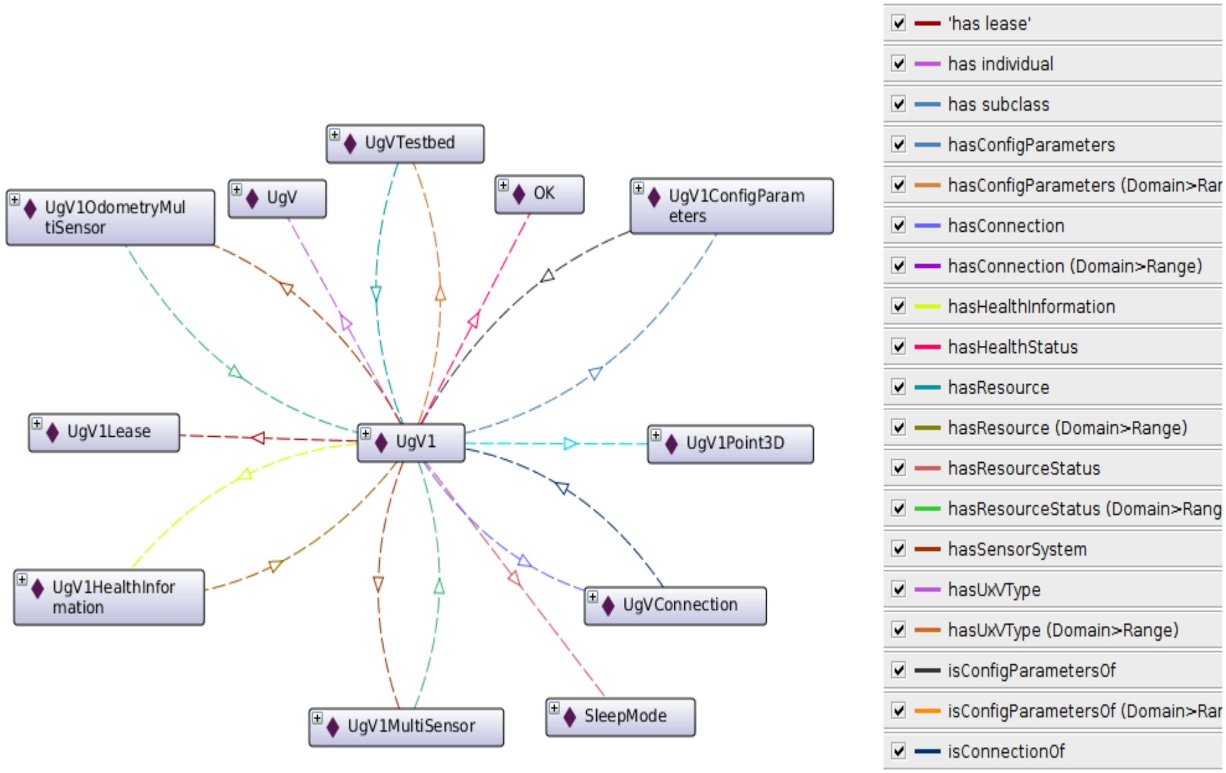

Figure 5: UxV Example

OMN UxV ontology uses predefined concepts (classes) and links (properties) from OMN ontology suite. OMN Federation ontology is used for the description of testbed. OMN Resource ontology is used for the description of UxV resources. OMN Lifecycle is used for the reservation process of UxVs. OMN Wireless ontology is used for the description of UxV communication capabilities. Finally, for the location of RAWFIE testbeds and UxVs Geo RSS Feature Model and ontology [18] is used.

4.2.1.2 OMN Sensor Ontology

The OMN Sensor ontology describes the attached sensors of RAWFIE resources and the sensors record measurements for a variety of phenomena. It focuses on the sensor characteristics that are involved in the selection of the appropriate UxV. Thus, as interesting features are considered the following:

- Feature of Interest (Air, Ground, Water)
- Measured Property (Temperature, Velocity, Pressure, Electric Current Rate, etc.)
- Unit of measured property.
- Sensor description (vendor name, product name, serial number, description).

For the description of the sensors we used the Semantic Sensor Networks (SSN) ontology, which is developed by the W3C Semantic Sensor Networks Incubator Group (SSN-XG) [19], and ontology for quantity kinds and units [20]. Figure 6 depicts the structure of OMN Sensor ontology.

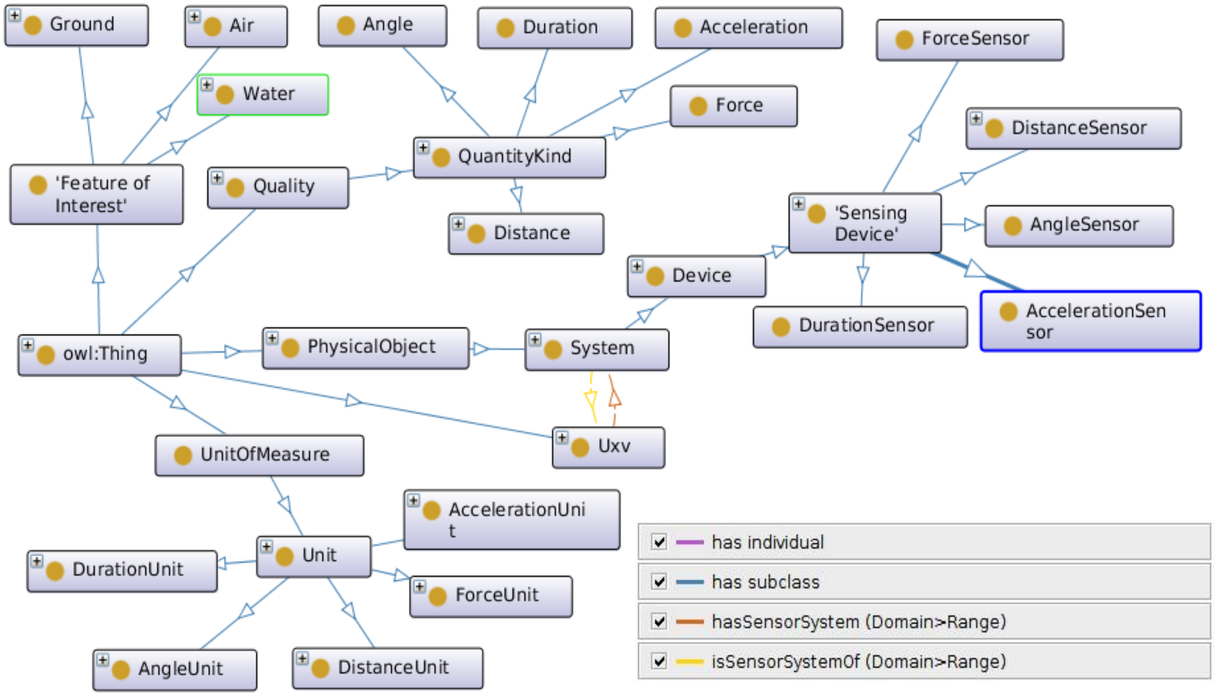

Figure 6: OMN Sensor Ontology

The set of sensors of each UxV is described by the ssn: System class. System class is linked with the UxV class of the OMN UxV ontology. All basic sensors of the System are described by the corresponding subclass of the ssn: sensing Deviceclass. The measuring property of each sensor is represented by the qu: QuantityKind class (property) and its subclasses. These classes are linked with the ssn: Feature Of Interest class that define if the property corresponds to “Air”, “Ground” or “Water” environment.

Figure 7 depicts the description of a sensor system attached on the ground unmanned vehicle (UgV) namedUgV1. This sensor system (UgV1MultiSensor) is equipped with an odometry (UgV1OdometryMultiSensor) and a laser sensor (UgV1LaserSensor). UgV1 Odometry Multi Sensor individual contains the basic sensors for measuring velocity (Ugv1OdometryVelocityorSpeed Sensor) and rotational speed (Ugv1OdometryRotationalSpeedSensor) respectively. The velocity sensor is linked with 'metre per second' individual and velocity individual (observing property).The rotational speed sensor is linked with 'radian per-second' individual and ‘normal rotational speed’ individual (observing property). UgV1 Laser Sensor individual is connected with 'metre' unit and distance (observing property) individuals. The ‘normal rotational speed’, velocity and distance properties are linked with the ground individual of the Feature of Interest class.

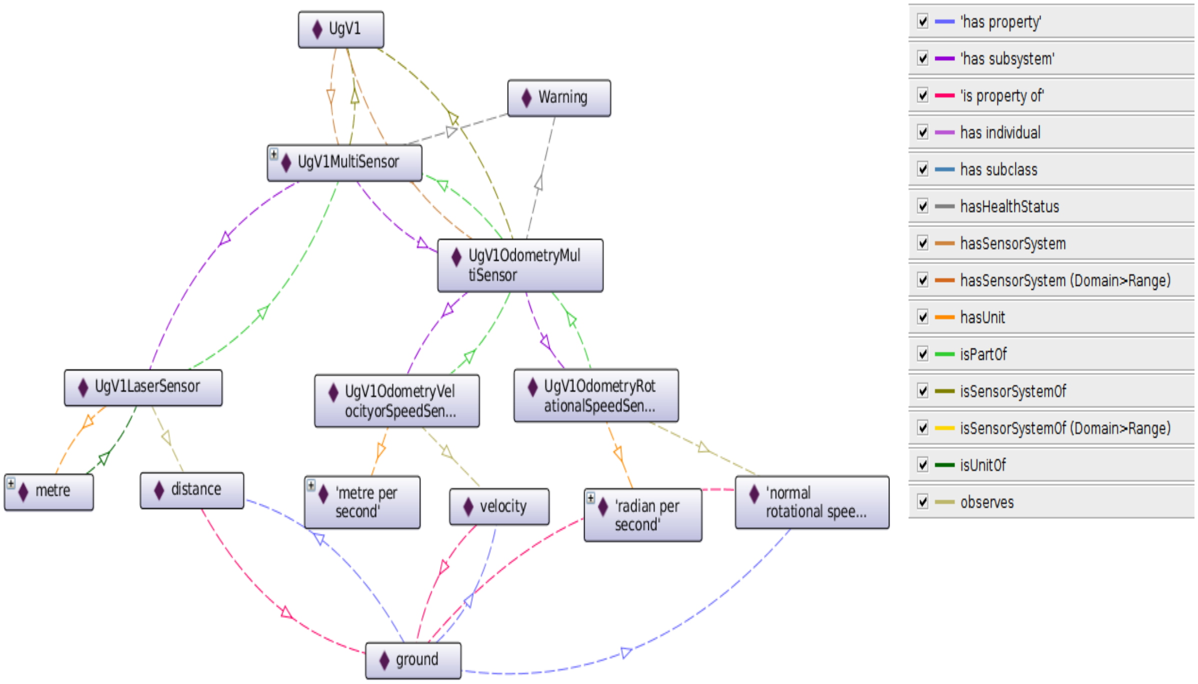

Figure 7: UgV Sensor System Example

4.3 Standards

4.3.1 Data Analytics

Here, we will mention standards that can be used for describing the results of the data analytical process. SparkML supports the export of models in PMML [2] (Predictive Model Markup Language to describe the generated models) provided that the ones, that we will generate, are covered by the current PMML version (v 4.2) [1]. While we will not be explicitly providing PMML files for our models (as per paragraph 1) the user can freely export their models (or variations to our models) in PMML via Spark and the Apache Zeppelin interface. Briefly PMML is an industrial standard that is used for the exchange of machine learning and data mining models between different applications and data analytical environments. It is based on XML and offers support for models generated as a result of different data mining tasks such as association rule discovery, classification, regression and clustering. A description of a data mining model in PMML contains the following elements:

- a header which provides general information about the model such as the analytical environment that generated the model, generation timestamps etc.
- a data dictionary describing the dataset from which the model was generated
- a data transformation component describing transformations that are applied to the data prior to modelling, such as normalization, discretization
- the model component describing the learned model.

4.3.2 Geospatial Data

Geospatial data is stored and processed in various, quite diverse, formats. Internally a common representation of the geospatial data will be used, which really simplifies the data handling. This representation hasn't been decided yet. However, imported data comes in any of the mentioned formats of section 2.1.4, or even another different one.

A list of common formats and standard is in the table below. Many of the standards are from the OGC [6].

Format	Description
Shapefile [3]	- de factor standard (designed by ESRI) to store vector data - supported by almost all GIS systems - only one geometry type per Shapefile - consists of multiple files - attribute data stored in dBASE (version IV) database (.dbf file) [4]
GeoPackage [5]	- recently developed OGC standard to store all kinds of geospatial related data (vector features, tile matrix sets of imagery and raster maps at various scales , schema, metadata) - database file that can be accessed and updated directly without intermediate format translations - can be seen as a modern replacement for shapefiles with the following advantages: ○ only one file instead of multiple files ○ smaller file sizes ○ wider spectrum of attribute types ○ less constraints (e.g. length of attribute names)

GML [6]	- *Geography Markup Language* - OGC standard to exchange vector data via XML files - very flexible and adaptable to individual needs - used in many open source systems
KML [7]	- *Keyhole Markup Language* - OGC standard to exchange vector data via XML files - mainly used by Google Earth
WMS [8]	- *Web Map Service* - OGC standard protocol for serving geo-referenced map images (raster data) - images are generally generated by a map server (most using data from a GIS database)
WMTS [12]	- *Web Map Tile Service* - OGC standard protocol for serving geo-referenced raster data - very similar to WMS, but with much simpler request interfaces - the raster data provided is normally pre-calculated and hence the server-side computing time is very low, making WMTS a very fast and responsive service
WFS [9]	- *Web Feature Service* - OGC standard protocol which provides an interface for geographical feature requests (vector data)
World-File [10]	- de factor standard (designed by ESRI) to store raster data - supported by almost all GIS systems - a text file (in conjunction with an picture file) that describes the projection of a picture into a specific coordinate system
GeoTiff [11]	- public domain metadata standard - allows geo-location information to be embedded within a Tiff image file

Also many other formats exist (structured text files, e.g. formatted as CSV, JSON (GeoJSON) or XML as well as many proprietary binary formats) that are used to store geospatial data.

5 Data Sharing, Archiving and Preservation

RAWFIE will consider all the necessary procedures for archiving and provision of long-term preservation of either the experimental data or any of the available data through the Open Access RAWFIE outcomes.

5.1 Data sharing

A specific dissemination strategy will be developed in RAWFIE in parallel with the implementation activities, with the aim to keep potentially interested stakeholders informed about the availability and the possibility, in case they conduct an experiment, to have access to their experimental data or any kind of data described in Section 3.

The strategy used for the dissemination of research data will include:

- identification of the different type of stakeholders (users or groups of users) that will be the intended "recipients" of the dissemination
- identification of the most suitable tools or mechanisms to be used for the dissemination, according to the type of audience
- implementation/use of the above mentioned dissemination tools or mechanisms

Possible stakeholders interested in the data generated by the project are:

- Experimenters
- Universities and research institutes
- UxVs and, in general, technology manufacturers (e.g. sensor or wireless communication solutions providers)
- Owners of institutional repositories (if any)

5.1.1 Sensor data from experiments

Sensors data are distributed via the Kafka message bus inside the RAWFIE system and are persistently stored in a central database (access only to trusted RAWFIE components). They are made available to the experimenter via the Visualisation and the Data Analysis Tools. Furthermore, each experimenter will have access to its experiments' raw data.

5.1.2 Data analysis results

The Data Analysis Tool uses the Graphite[32] framework to visualise sensor values and the analysis results.

[32] http://graphiteapp.org

5.1.3 Exploitation

A number of business cases are briefly described below:

- **Patenting:** UxV manufacturers can patent several components raised by the needs of RAWFIE experimentation environment like a redundant propulsion system for UAVs.
- **Model valorisation**: Since there is not much direct business to be made out of the models themselves, the valorisation of the models will probably be done through the use of the RAWFIE platform. For that matter, please refer to WP2 deliverables.
- **Data valorisation**: The collected data has an explicit and an implicit value. The explicit value lies in the information that can be extracted from it, after analysis and interpretation, e.g. for tuning or debugging the RAWFIE platform, its component or for adjusting the parameters of the resources, such as UxVs or Testbeds; this value can be commercially traded, as it will be the case for the data obtained from any other experiment. The implicit value comes from the nature of this data, which can, then, be used as reference data for other UxV or testbed owners that would like to introduce their assets or technologies into the RAWFIE infrastructure for later being used as a resource or a service; this value is probably not directly exploitable.

5.2 Staging processing for experiment validation streams

The RAWFIE architecture follows the principle of informed consent by end-users. Participants to the experiments will be required to have previously given their consent to take part in it, with a clear understanding of what the collected data are and what is their potential use and distribution. In order for a user to access the gathered information, he will have to register to a dedicated service. During this process, the end-user will be notified about the purpose and the scope of the project via a “Terms and Conditions” notice.

Special cases demand for specific disclosure process (clearance) and terms of use. This is for example the case of testbeds close to sensitive areas. Such is the case for the following test-bed:

– Since Testbed Boundaries are lying within the Naval Fortress of Skaramanga and due to the proximity of Salamis Naval Base in order to prevent any sensitive information leakage with respect to operational capabilities all data collected by any means through UXVs will be submitted to thoroughly check by a Hellenic Navy Intelligence “Safety committee” prior public release. Sensitive data might be censored in order to fulfill above-mentioned restrictions.

Specific Sensor Restrictions will be disseminated to testbed users/experimenters upon commission of UxVs and after sensors capabilities have been notified. Furthermore, any data which not falls into prior restriction should be handled by testbed operators with respect to EU and national laws concerning Personal Information privacy.

5.3 Data Archiving and preservation

RAWFIE will consider all the necessary procedures for archiving and provision of long term preservation. Suitable file formats and appropriate processes for organizing files will be followed. In organizing the different data files the following steps could be considered:

- File version control
- File structure
- Directory structure and file naming conventions

RAWFIE different repositories are:

- ***Master Data Repository*** to contain all the management data sets (experiments, EDL scripts, bookings, testbeds and resources, status information of testbeds and their resources, and so on) of RAWFIE. PostgreSQL [6] with PostGIS extension was chosen for the implementation, as it is well supported, open source and stable, and to be able to easily handle geo-referenced data.
- ***Measurements Repository*** that will use a big data storage system for storing the large number of measurements that will be coming from the sensors on board of the UxVs during the experiments. The popular big data solution "Hadoop Distributed File System" [13] is one of the potential solutions for this purpose, however the specific technological choice will be detailed in further WP4 deliverables and WP5. In addition, a NoSQL solution is expected to be adopted in the 2nd implementation iteration to better manage the data sets. Currently HBase (running on top of HDFS) has been identified for this purpose. HBase supports random, real-time read/write access with a goal of hosting very large tables atop clusters of commodity hardware. HBase features include i) consistent reads and writes, ii) automatic and configurable sharding of tables and iii) automatic failover support. Hbase can be connected with Apache Confluent/Kafka and can use ZooKeeper for coordination of "truth" across the cluster. As region servers come online, they register themselves with ZooKeeper as members of the cluster. Region servers have shards of data (partitions of a database table) called "regions". This supports the online streaming of raw data generated by experimenters with no delays at reads and writes. For further interpretation of the raw data analysis results repository is used.
- ***Analysis Results Repository*** uses a seperated database for performing the Data Analytics task over the results of the experiments. The Graphite data analysis framework will be used with the database called Whisper [14].
- ***Users & Rights Repository*** uses a LDAP [15] repository, as the LDAP is the de facto standard for user management. It stores all user related data (name, organisation, address, password) and group memberships (roles based access control). The selected implementation is OpenDJ [16].

Except for the Analysis Results Repository, all used repository systems (PostgreSQL, HDFS, OpenDJ) support replication, thus, they do provide fault tolerance. In case of data loss in the

Analysis Results Repository, they can be recomputed using data stored in the Measurements Repository.

In addition for the long-term access appropriate data documentation will be provided. Full understanding and analysis of the metadata that may be needed will be considered. For instance, for improving documentation process we could classify the metadata in two levels: project- and data-level. Project-level metadata describes the "who, what, where, when, how and why" of the dataset, which provides context for understanding why the data were collected and how they were used.
Examples of project-level metadata:

- Name of the project
- Dataset title
- Project description
- Dataset abstract
- Principal investigator and collaborators
- Contact information

Dataset level metadata are more granular. They explain, in much better detail, the data and dataset.
Examples of data-level metadata:

- Data origin, experimental, observational, raw or processes, models, images, etc.
- Data type: integer, boolean, character, floating, etc
- Data acquisition details: sensor deployment methods, experimental design, sensor calibration methods, etc
- File types: CSV, mat, tiff, xlsx, HDF
- Data processing methods
- Dataset parameter list: Variable names, Description of each variable, units

The external repositories that can be used for the purposes of archiving and long-term storage were described above (see Section 5.3). These repositories are open therefore there will not add expenses for the RAWFIE consortium. In case of additional procedures are needed for the long-term maintenance the project consortium will cover the respective costs.

6 References

[1] PMML 4.2: http://www.dmg.org/pmml-v4-2.html

[2] PMML: An Open Standard for Sharing Models,Alex Guazzelli, Michael Zeller, Wen-Ching Lin and Graham Williams,The R Journal, Volume 1/1, May 2009.

[3] ESRI Shapefile Technical Description**,** ESRI, July 1998, http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf

[4] http://www.dbase.com/

[5] OGC GeoPackage Encoding Standard, Paul Daisey, version: 1.0.1, April 2015, http://www.geopackage.org/spec/

[6] Geography Markup Language, OGC, various versions, http://www.opengeospatial.org/standards/gml

[7] KML, OGC, various versions, http://www.opengeospatial.org/standards/kml/

[8] Web Map Service, OGC, various versions, http://www.opengeospatial.org/standards/wms/

[9] Web Feature Service, OGC, various versions, http://www.opengeospatial.org/standards/wfs/

[10] About world files, ESRI, http://webhelp.esri.com/arcims/9.2/general/topics/author_world_files.htm

[11] GeoTIFF Format Specification, Niles Ritter, version 1.8.2, December 2000, http://www.remotesensing.org/geotiff/spec/geotiffhome.html

[12] Web Map Tile Service, OGC, various versions, http://www.opengeospatial.org/standards/wmts

[13] http://hadoop.apache.org/index.html

[14] http://graphite.readthedocs.io/en/latest/whisper.html

[15] https://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol

[16] https://forgerock.org/opendj/

[17] A. Willner, C. Papagianni, M. Giatili, P. Grosso, M. Morsey, Al-Hazmi Y., I. Baldin, "The Open-Multinet Upper Ontology - Towards the Semantic-based Management of Federated Infrastructures", The 10th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities (TRIDENTCOM 2015), Vancouver, Canada, June 2015.

[18] Lieberman J., Signh R., Goad C., W3C Geospatial Vocabulary, Available at: https://www.w3.org/2005/Incubator/geo/XGR-geo-20071023/

[19] Compton, M., Barnaghi, P., Bermudez, L., GarcíA-Castro, R., Corcho, O., Cox, S., & Huang, V. (2012). “The SSN ontology of the W3C semantic sensor network incubator group”, Web Semantics: Science, Services and Agents on the World Wide Web, 17, 25-32.

[20] Lefort L, “Ontology for quantity kinds and units: units and quantities definitions”, W3 Semantic Sensor Network Incubator Activity, 2005.

A ANNEX I

SUMMARY TABLE 1

FAIR Data Management

This table provides a summary of the Data Management Plan (DMP) issues to be addressed during RAWFIE lifetime.

DMP component	Issues to be addressed	Related Sections
1. Data summary	• State the purpose of the data collection/generation • Explain the relation to the objectives of the project • Specify the types and formats of data generated/collected • Specify if existing data is being re-used (if any) • Specify the origin of the data • State the expected size of the data (if known) • Outline the data utility: to whom will it be useful	**D7.5 – Section 2 Dataset description and processing**
2. FAIR Data 2.1. Making data findable, including provisions for metadata	• Outline the discoverability of data (metadata provision) • Outline the identifiability of data and refer to standard identification mechanism. Do you make use of persistent and unique identifiers such as Digital Object Identifiers? • Outline naming conventions used • Outline the approach towards search keyword • Outline the approach for clear versioning • Specify standards for metadata creation (if any). If there are no standards in your	**D7.5 - Section 3 and Section 4**

	discipline describe what type of metadata will be created and how	
2.2 Making data openly accessible	• Specify which data will be made openly available? If some data is kept closed provide rationale for doing so • Specify how the data will be made available • Specify what methods or software tools are needed to access the data? Is documentation about the software needed to access the data included? Is it possible to include the relevant software (e.g. in open source code)? • Specify where the data and associated metadata, documentation and code are deposited • Specify how access will be provided in case there are any restrictions	**D7.5 - Section 3 and Section 4**
2.3. Making data interoperable	• Assess the interoperability of your data. Specify what data and metadata vocabularies, standards or methodologies you will follow to facilitate interoperability. • Specify whether you will be using standard vocabulary for all data types present in your data set, to allow inter-disciplinary interoperability? If not, will you provide mapping to more commonly used ontologies?	**D7.5 - Section 4.2** **Research on progress for further interoperability of the data compoennts.** **Desctiption in D7.5(c)**
2.4. Increase data re-use (through clarifying licences)	• Specify how the data will be licenced to permit the widest reuse possible • Specify when the data will be made available for re-use. If applicable, specify why and for what period a data embargo is needed • Specify whether the data produced and/or used in the project is useable by third parties, in particular after the end of the project? If the re-use of some data is restricted, explain why • Describe data quality assurance processes • Specify the length of time for which the data	**TBA in the version 3 – D7.6**

	will remain re-usable	
3. Allocation of resources	• Estimate the costs for making your data FAIR. Describe how you intend to cover these costs • Clearly identify responsibilities for data management in your project • Describe costs and potential value of long term preservation	**TBA in the version 3 – D7.6**
4. Data security	• Address data recovery as well as secure storage and transfer of sensitive data	**Section 5 and 6**
5. Ethical aspects	• To be covered in the context of the ethics review, ethics section of DoA and ethics deliverables. Include references and related technical aspects if not covered by the former	**Deliverable D1.13**
6. Other	• Refer to other national/funder/sectorial/departmental procedures for data management that you are using (if any)	**TBA in the version 3 – D7.6**