This project has received funding from "HORIZON 2020" the European Union's Framework Programme for research, technological development and demonstration under grant agreement no 645220

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	**D7.4 Data Management Plan (a)**		
Confidentiality	PU	**Deliverable type**[1]	R
Deliverable File	D7.4 Data Management Plan (a)	**Date**	30.06.2015
Approval Status[2]	2nd Reviewer	**Version**	1.0
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
Blerina Lika	UoA	b.lika@di.uoa.gr
Kostas Kolomvatsos	UoA	kostasks@di.uoa.gr
Sarantis Paskalis	UoA	paskalis@di.uoa.gr
Philippe Dallemagne	CSEM	pda@csem.ch
Marcel Heckel	Fraunhofer	Marcel.Heckel@ivi.fraunhofer.de
Patrick Brausewetter	Fraunhofer	patrick.brausewetter@ivi.fraunhofer.de
Alexandros Kalousis	HES-SO	Alexandros.Kalousis@unige.ch
Savvas Chatzichristofis	CERTH	savvash@gmail.com
Richardo Martins	MST	rasm@oceanscan-mst.com
Alexander Sousa	MST	alex@oceanscan-mst.com
Miquel Cantero	Robotnik	mcantero@robotnik.es
Rafa Lopez	Robotnik	rlopez@robotnik.es

REVIEWERS TABLE

Name	Company	E-Mail
Giovanni Tusa	IES	g.tusa@iessolutions.eu
Nikolaos Priggouris	HAI	PRIGGOURIS.Nikolaos@haicorp.com

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	2015-05-04	First Document Issue with Introduction and ToC	All
0.2	2015-05-15	Structure updated /assignments among partners	Sections 2,3 4
0.3	2015-05-19	Regulations for data sharing and repositories	Section 4
0.4	2015-06-05	Updates on data references and formats	Sections 2.1, 2.3, 2.4, 3.1,3.3, 3.4
0.5	2015 – 06-18	Additions and comments addressed Archiving procedures were added	Sections 2.2, 2.3, 4
0.6	2015 – 06 - 19	Scope and doc introduction were enhanced	Section 1
0.7	2015 – 06 23	MST, Robotnik and IES provided their updated contributions	Sections 2.4, 2.5, 3.2.1. 3.2.2
0.8	2015 – 06-24	The internal review process starts	All
0.9	2015 – 06 - 26	Comments raised by reviewers	All
1.0	2015 – 06 - 30	The comments addressed by the involved partners. The deliverable was prepared for the submission.	All

Abstract:

The deliverable provides a first version of the data management plan that will be adopted by the RAWFIE project. The plan which follow the Horizon 2020 Work Program 2014-15 directives for Data Management Plan (DMP). The purpose of this deliverable is to support the data management life cycle for all data that will be collected, processed or generated by the project. It will provide an outline of the data types the project will generate, whether and how it will be exploited or made accessible for verification and re-use, and how it will be curated and preserved.

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

Table of Contents

1 Introduction 8
1.1 Scope of D7.4 8
1.2 Abbreviations 9
2 Data set reference, standards and metadata 10
2.1 Raw data and sensor observations 10
2.2 Processed data, models and analytics 12
2.3 Geospatial data 13
2.4 Image data from UxV platforms 16
2.4.1 Robotnik platforms 16
2.4.2 MST platforms 17
2.5 Simulated data 17
2.5.1 Robotnik platforms 17
2.5.2 MST platforms 19
3 Data sharing 21
3.1 Data access procedures 21
3.2 Mechanisms for dissemination and sharing research data 21
3.2.1 Mechanisms for dissemination 22
3.2.2 Software tools for sharing research data 23
3.2.3 Repository concept for enabling re-use 24
4 Archiving and preservation 25
References 26

Part II

List of Figures

Figure 1: Data Management Cycle

List of Tables

Table 1: Abbreviations
Table 2: Sensor observation data and metadataTable 1: Abbreviations
Table 3: Geospatial data overview
Table 3: Geospatial data overview

Part III: Main Section

1 Introduction

1.1 Scope of D7.4

The purpose of "D7.4 - Data Management plan" is to provide an overview of the main elements of the data management policy that will be used by the with regard to all datasets that will be generated by the project. It also describes the access granted to all parties interested in the data generated by the RAWFIE system during its development, tests and operations. Finally it discusses the compliance of the RAWFIE data structure, management and policy with respect to EU regulations and directives. This deliverable will be evolved and updated during the lifespan of the project.

Figure 1 presents the steps and actions involved in a typical data management cycle.

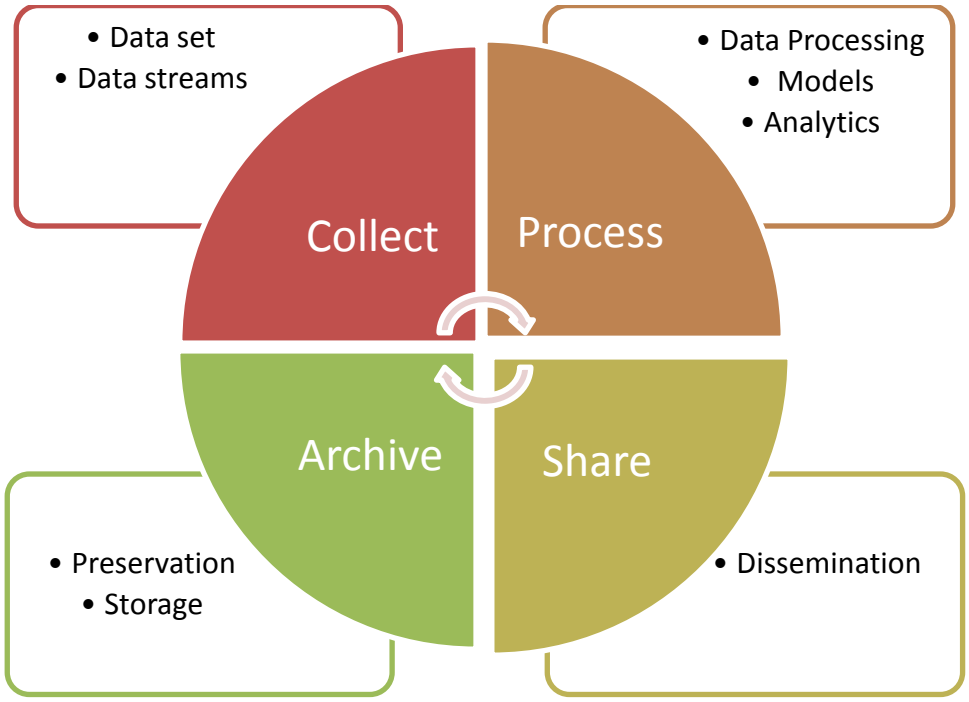

Figure 1: Data Management Cycle

This document is structured as follows: Section 2 describes the data types and metadata that will be collected and processed during the experimentation and the project lifetime as well as the respective standards and formats. Section 3 contains the data access procedure and the dissemination mechanisms that will take place to provide reusability and access in the future. Finally, Section 4 describes the procedures for the archiving and long-term storage.

1.2 Abbreviations

Abbreviation	Meaning
DMP	Data Management Plan
SOS	Sensor Observation Service
OGC	Open Geospatial Consortium
GML	Geography Markup Language
KML	KML - formerly Keyhole Markup Language
WMS	Web Map Service
WMTS	Web Map Tile Service
WFS	Web Feature Service
CSV	Comma-separated values
JSON	JavaScript Object Notation
GIS	Geographic Information System
PMML	Predictive Model Markup Language
EDL	Experiment Description Language
XML	Extensible Markup Language
SensorML	Sensor Model Language
O&M	Observations and Measurements
SPS	Sensor Planning Service
SWE	Sensor Web Enablement
ML	Machine Learning
DM	Data Management
IMC	Inter Module Communication
ROS	Robot Operating System
LLF	LSTS Log Format
ISO	International Organization for Standardization

Table 1: Abbreviations

2 Data set reference, standards and metadata

This section will describe identifiers for the data set to be produced. It will include:

- description of the data that will be generated or collected
- its origin (in case it is collected)
- nature and scale, and to whom it could be useful
- whether it underpins a scientific publication
- information on the existence (or not) of similar data and the possibilities for integration and reuse.

The following sections include the potential data types to be generated by the project and reference to existing suitable standards of the discipline.

2.1 Raw data and sensor observations

Raw sensor data will be collected by on-board mobile sensors located in the UxV devices. These data has not been subjected to processing or any other manipulation. The types of raw data will be related to the different sensor types that will be used in the context of the RAWFIE project. We can classify the sensors and the relevant data into the following categories:

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

All these sensors form the basis for calculating the experiment context. The context and the metadata that could be retrieved are based on the relevant format and standard. The basic information that will be generated by the RAWFIE sensors is specified as follows in Table 2:

Data	Data type	Description
Identifier (ID)	String	Unique identification of the sensor
Owner	String	Any text that describes the owner/manufacturer of the sensor
Sensor type	String	Description of the sensor type
Observed area	Coordinates (latitude, longitude, elevation)	The geographical area within which the associated observations were made
Phenomenon	String	The phenomenon description (e.g., temperature)
Observed result	Integer or String	The observed value that is related to the phenomenon

Unit of measurement	String or Character	The unit of measured phenomenon (e.g., degree Celsius °C)
Date and time	Timestamp	Sequence of characters specifying when the observation took place
Offering ID	String	Text description that can be used for group purposes

Table 2: Sensor observation data and metadata

There are already some similar initiatives and EU projects that have generated and collected sensor observations, but for different purposes and applications. For instance Fed4FIRE project has collected such data (e.g., environmental data) from different experiment executions. These data are strictly related to the experimentation scenarios that could be a combination of different data types (e.g., sensor data and geospatial data) therefore there is no possibility to integrate the existing sensor observation data for RAWFIE purposes. The project will collect and generate such data from its own experiment executions.

Standards

There are already some standards and formats to encapsulate and manage the information from sensor observations and raw data. The sensor observation standard that could be adopted by RAWFIE is yet to be decided. However, some existing standards could be:

- Sensor Observation Service (SOS) [1]

The SOS is an interface provided by the OGC consortium in order to allow access and distribution for sensor description and observation. The specification leverages the Observations and Measurements (O&M) specification to encode observations and the Sensor Model Language (SensorML) specification to encode sensor descriptions. Both of these formats are based on Extensible Markup Language (XML). This standard defines a Web-based interface (Web Service) that allows querying observations, sensor metadata or representations of observed features. Further, it provides means to register new sensors or remove existing ones. It also defines operations to insert new sensor observations. The SOS operations follow the general pattern of other OGC Web Services and inherit or re-use, when needed, elements defined previously.

- Sensor Model Language (SensorML) [2]

SensorML provides models and XML encodings for describing any process related to sensor system. Processes, described in SensorML, define their inputs, outputs, parameters, method and they also provide relevant metadata. This standard includes sensors and actuators as well as computational processes applied pre- and post-measurement. The main objective is to enable interoperability, first at the syntactic level and later at the semantic level (by using ontologies and semantic mediation), so that sensors and processes can be better understood by machines, utilised automatically in complex workflows, and easily shared between intelligent sensor web nodes. This standard is one of several implementation standards produced under OGC's Sensor

Web Enablement (SWE) activity. In RAWFIE, SensorML can be used to describe different types of sensors (e.g. environmental, air pressure).

- Observations and Measurements (O&M) [3]

O&M defines a conceptual schema encoding for observations and features involved in sampling when making observations. O&M provides document models for the exchange of information describing observation acts and their results, both within and between different scientific and technical communities. This encoding is an essential dependency for the OGC Sensor Observation Service (SOS) Interface Standard. This standard can be leveraged by RAWFIE to describe the measurements derived by different sensor types.

- Sensor Planning Service (SPS) [4]

The SPS is an interface standard defining interfaces for queries that provide information about the capabilities of a sensor and how to task the sensor. The standard is designed to support queries that have the following purposes: a) to determine the feasibility of a sensor planning request, b) to submit and reserve/commit such a request, c) to inquire about the status of such a request, d) to update or cancel such a request; and e) to request information about other OGC Web services that provide access to the data collected by the requested task.

2.2 Processed data, models and analytics

Processed data refer to the models and the statistics that will be generated from the stream analytics platform. Typical models include classification and outlier detection models. The data mining and machine learning community traditionally relies on the exchange and publication of datasets. This is achieved by a number of relevant data repositories which will be described in a later section, and much less through models. Nevertheless there is a standard, PMML [5] which allows the description and sharing of learned models between different analytical environments. 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.

As mentioned in the previous paragraph the learning and mining community has focused on the exchange of data and not on that of models. Once the raw data for a given learning tasks are available, different teams can test their own approaches and 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.

Standards

Here we will mention standards that can be used for describing the results of the data analytical process. We will consider the use of PMML [7], 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) [5]. 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 modeling, such as normalization, discretization
- the model component describing the learned model.

2.3 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). Basically geometry data can be distinguished into vector and raster data.

Vector data means that entities consist of one or more coordinates that form a geometric primitive (geometry type). The commonly supported geometric primitives are points, multi-points, lines, multi-lines, polygons and multi-polygons. The difference between the simple and the multi geometries is that one multi geometry consists of one or many simple geometries of the same type. There are also geometry collections consisting of geometries of different types, but they are not very commonly used. Apart from that specific vector formats may also support ellipses, splines, etc.

Raster data means that an entity is a picture (of one or more sub entities) in conjunction with a definition of a projection into a specific coordinate system specifying the exact extent of this picture in terms of the coordinate reference system. Using this definition, the picture could be rendered on the right place and shaped in a map using this reference system.

The following list of geospatial information, gives an overview of the types of data with a spatial reference that will be possibly generated and / or collected inside RAWFIE.

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	An 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, see also section **Error! eference source not found.**
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.

Table 3: Geospatial data overview

During the implementation of the project new kinds of geospatial data will possibly be examined as new services will be integrated that have not been foreseen in the forefront of the project. When this happens, this list will be updated to reflect the new data types.

Standards

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 is yet to be decided. However, imported data comes in any of the mentioned formats, or even another different one.

A list of common formats and standard is in the table blow. Many of the standards are from the OGC [8].

Format	Description
Shapefile [9]	- 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) [10]

GeoPackage [11]	- 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 [12]	- *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 [13]	- *Keyhole Markup Language* - OGC standard to exchange vector data via XML files - mainly used by Google Earth
WMS [14]	- *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 [18]	- *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 [15]	- *Web Feature Service* - OGC standard protocol which provides an interface for geographical feature requests (vector data)
World-File [16]	- 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 [17]	- public domain metadata standard - allows geo-location information to be embedded within a Tiff image file

Table 4: Geospatial data formats and standards

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.

2.4 Image data from UxV platforms

2.4.1 Robotnik platforms

Robotnik's platforms make use of ROS (Robot Operating System) standard formatting. ROS image format message are sensor_msgs and images.

There are three standard data types to get video images in ROS:

- "raw": The default transport
- "compressed": JPEG or PNG image compression
- "theora": Streaming video using the Theora codec.

Normally, every camera driver is able to publish all of these formats, image_transport ROS package is the recommended tool to do it. Robotnik platforms are usually integrating camera sensors such as Microsoft Kinects or Asus XTION and AXIS PTZ Cameras. The last ones also provide its own RTSP Web Server streaming MPEG-4.

The ROS package sensor_msgs defines messages for commonly used sensors, including cameras and scanning laser rangefinders. A lot of data can be found under ROS as many other applications and programs make use of such data. For an example on how to work with specialized ROS messages please refer to MathWorks [19].

As stated with another critical components/software of these platforms, the communication with non-ROS systems shall be done by using tools such as the Rosbridge server (see D4.1) which involve JSON libraries. In addition, regarding Image data exchange, a better way to get the video stream is by using the package mjpeg_server that connects to the ROS topic and publishes the video stream via MJPEG server. Some other tools like *cv_bridge (*http://wiki.ros.org/cv_bridge) can be used to interface ROS and OpenCV by converting ROS images into OpenCV images, and vice versa.

2.4.2 MST platforms

Images and related data collected by MST vehicles are provided to the user as is, which means that the data format is a characteristic of the sensor/camera being used. As of the date of this writing the MST vehicles use two different types of cameras. The first type is an industrial camera that records video as individual JPEG images, the user may configure the desired frame rate which can be between 4 and 15 frames per second. Each JPEG image contains embedded navigation data (position and pose) encoded using the Exif format and the maximum available image resolution is 1376x1032 pixels. This type of camera is usually used to take georeferenced pictures of the seabed under low-light conditions occasionally with the aid of an external synchronized illumination module.

The second type of camera is capable of recording and streaming video encoded with MPEG-4 (H.264) at a frame rate of 30 frames per second and a resolution of 1280x720 pixels. This camera is normally used without the aid of artificial illumination and the video collected is commonly streamed in real time to a control center for surveillance purposes.

Standards

Both types of cameras used by MST comply with the following industry standards:

- ISO/IEC 10918-1:1994, Information technology - Digital compression and coding of continuous-tone still images: Requirements and guidelines.
- CIPA DC-008-2012, Exchangeable image file format for digital still cameras: Exif Version 2.3
- ISO/IEC 14496 MPEG-4 Standard Parts 1 to 31

2.5 Simulated data

This section contains the simulated data and formats that Robotnik's and MST's platforms uses during the experiment simulations, which possibly will be leveraged by the RAWFIE project.

2.5.1 Robotnik platforms

Robotnik platforms simulations are normally being carried by Gazebo Simulator [20]. Although it is not the only simulating software compatible with ROS, it is the most commonly used tool by ROS users because of its good integration with the platform. The main reason is that Gazebo is capable of simulating a wide range of testbeds, from a mapped indoor office, to outdoor environments with only a map and the model of the robot.

To summarize, this model structures is as follows:

- *Database*
  - *database.config* : Meta data about the database. This is now populated automatically from CMakeLists.txt
  - *model_1* : A directory for model_1
    - *model.config* : Meta-data about model_1
    - *model.sdf* : SDF description of the model
    - *meshes* : A directory for all COLLADA and STL files
    - *materials* : A directory which should only contain the textures and scripts subdirectories
      - *textures* : A directory for image files (jpg, png, etc).
      - *scripts* : A directory for OGRE material scripts
    - *plugins*: A directory for plugin source and header files

Robotnik has developed these models for its platforms and there's no need of rebuilding them. The key of these simulations lay on the information that Gazebo is processing. In order to carry these simulations, similar nodes (processes) to the ones that control the real robot are launched. The real difference relies on who is subscribing and publishing the data. For instance, Gazebo generates data such as raw images or joint movements for the robot to process, while translates movement commands generated by the robot to move the model in the simulator.

Standards

All the simulated data has the standard ROS format [21] that defines messages for commonly used sensors, including cameras and scanning laser rangefinders. An example of this format is shown below.

- Maintainer status: maintained
- Maintainer: Tully Foote <tfoote AT osrfoundation DOT org>
- Author:
- License: BSD

In addition Robotnik simulations follows geometry_msgs [22] format that provides messages for common geometric primitives such as points, vectors, and poses. These primitives are designed to provide a common data type and facilitate interoperability throughout the system. An example of geometry_msqs is shown below.

- Maintainer status: maintained
- Maintainer: Tully Foote <tfoote AT osrfoundation DOT org>
- Author: Tully Foote
- License: BSD

2.5.2 MST platforms

MST's simulated and real vehicles use the same data format for inter-module communication and for data storage to persistent media. This allows for simulated vehicles to use and replay data from real missions (e.g., environmental data, bathymetry). In doing so, while the vehicle's kinematics are simulated using the vehicle's model and physical characteristics, environmental sensor data can be simulated or taken from values collected in the field.

Standards

The data format in question is called IMC (Inter Module Communication) and comprises different logical message groups for networked vehicle and sensor operations. It defines an infrastructure that is modular and provides different layers for control and sensing. IMC defines the message entity as having an associated uniquely identifying number and consisting of a (possibly empty) sequence of data fields capable of representing fixed-width integers, floating point numbers, variable length byte sequences and inline messages (messages within messages). Integers can be signed or unsigned with sizes ranging from 8 to 64 bits. Floating point numbers have two sizes: 32 and 64 bits. Messages are prefixed with a header and suffixed with a footer to form a packet. Header and footer entities are defined as non-empty sequences of data fields and have the same structure for all packets.

In order to transmit a message or save it to persistent storage the message has to be encapsulated in a packet and serialized. Serialization is performed by translating the data fields of the packet entities (header, message and footer) to a binary stream in the same order as they were defined. The first field of the packet header is the synchronization number, used to mark the beginning of a packet and to denote its protocol version. By inspecting the synchronization number the recipient is able to deduce the byte order of the remaining data fields and perform the necessary conversions for correct interpretation. Using this approach, communication between nodes with the same byte order incurs in no byte order conversion overhead and communication between nodes with different byte orders only introduces the conversion overhead when deserializing packets. The complete IMC protocol is defined in a single eXtensible Markup Language (XML) document that, when changed, can be verified against a XML Schema (XSD). In the XML definition, each message field must have at least a name, one abbreviation (used for code generation) and a type. Optionally units and range of permissible values can also be defined.

Having a XML document describing the protocol has proven to be very practical for continuous development and testing. This happens because just after agreeing upon a specific version of IMC, two nodes can use the XML document to understand each other. Python and Java programs are used to automatically generate the IMC protocol reference documentation and optimized implementations exist in C++ and Java. Generating native code from the XML document has provided not only flexibility but also the performance needed for real-time execution in resource constrained computers.

In addition to the main serialization format described above, there are two complementary serialization formats with specific intents. One is the LLF (LSTS Log Format) format, a text format used for logging IMC, amenable to direct human understanding and easier to parse directly by many standard applications e.g. Matlab, Microsoft Excel, and custom mission review and analysis software. In order to be possible to review data from past missions, the LLF format had to be independent of the originating IMC protocol description (since the message format can change over time). Our approach was to define this tab-separated log format where, for each column (message field), there is a header describing its data type, name and units to be used when representing the data. Another format is the IMC-XML, which can be used as a simplified serialization format itself for inter-module interoperability. The main reason for this additional format is to enable the integration of web-based components and web-enabled third-party sensors into large-scale data dissemination applications.

3 Data sharing

This section describes the access procedures, the technical mechanisms for dissemination and necessary software for enabling re-use.

3.1 Data access procedures

In RAWFIE, data access will be specified with regards to the project phases, i.e. the project implementation period and the post-EC-funding phase. During the project implementation the data access will be set by default as public providing the possibility to the experimenters to set the access private according to his/her preferences. In this case, some experiments' data will be restricted and the results will not be shared to the public. The facilities will be open to experimenters, beyond participation in the Open Calls for Experiments, until the end of the project. Interested parties to conduct experiments on the RAWFIE facilities can contact the project consortium at any time to explore opportunities. After the project life-cycle, when the project's outcomes will be exploited for both commercial and public funded cases, a different policy will be followed. The data access will be set by default as private and only in specific cases as public according to the experimenters' preferences.

The experimenters and other stakeholders will have the opportunity to access the data through the project portal or the respective repository. The identification and the type of the repository (i.e., institutional, standard repository for the discipline) where the research data will be stored will be defined in an upcoming iteration of this deliverable.

Finally, RAWFIE project results will be fully complied with the EU Regulation and Directives. More specifically, the data generated by the project will follow the following Directives:

- Data Protection Directive, Directive 95/46/EC
- Data Retention Directive, Directive 2006/24/EC
- INSPIRE Directive Infrastructure for Spatial Information in the European Community, , Directive 2007/2/EC
- Marine Strategy Framework Directive, Directive 2008/56/EC
- Water Framework Directive, Directive 2000/60/EC

Possible integration into EC frameworks

The EC is supporting numerous initiatives, in particular for the consistent representation of common concepts, such as the building and city modeling, representation and parameters (IRCABC). RAWFIE may contribute to such initiatives by using or extending existing models.

3.2 Mechanisms for dissemination and sharing research data

This section outlines of technical mechanisms for dissemination and necessary software for enabling re-use.

3.2.1 Mechanisms for dissemination

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 to access new research data. Research data include experiments' results, as well as any other kind of data described in Section 2, generated within the platform during the execution of the experiments.

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 abovementioned 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 manufactures (e.g. sensor or wireless communication solutions providers)
- Owners of institutional repositories (if any)

Potential dissemination mechanisms, together with the description on how they could be used, and the possible mapping with the different stakeholder groups they could reach, are provided in

Dissemination mechanism	**How it will be used**	**Stakeholder type**
Project website	News will be published with information about executed experiments and data availability	• Experimenters • Universities and research institutes • UxVs and, in general, technology manufactures (e.g. sensor or wireless communication solutions providers) • Owners of institutional repositories (if any)
Newsletter / email	Periodic news sent to selected stakeholder groups, will be also used to share information about the availability of relevant research data	• Experimenters • Universities and research institutes

Publications	At certain points in time, results and statistics coming from the experiments will be published in scientific papers, together with the information on how to access them	• Experimenters • Universities and research institutes • UxVs and, in general, technology manufactures (e.g. sensor or wireless communication solutions providers) • Owners of institutional repositories (if any)
Public Web feeds	Feeds will be created where interested stakeholders can subscribe in order to receive notification about new available research data	• Experimenters • Universities and research institutes • UxVs and, in general, technology manufactures (e.g. sensor or wireless communication solutions providers) • Owners of institutional repositories (if any)
Social Media (e.g. Twitter, Facebook, Linkedin)	Notification about new available research data will be regularly published through the social media channels setup by the project	• Experimenters • Universities and research institutes • UxVs and, in general, technology manufactures (e.g. sensor or wireless communication solutions providers) • Owners of institutional repositories (if any)

Table 5: Mapping of mechanisms to stakeholder groups

3.2.2 Software tools for sharing research data

Further to the dissemination strategy that will be developed in order to inform interested stakeholders about availability of new research data, and to the repository concepts explained in the following section, a number of technical solutions will be taken into considerations, to allow the possibility to share some of the generated data in an almost real-time manner.

Software mechanisms or interfaces for sharing some of the data types mentioned in this document far include:

- Geospatial Data
  - WMS and WFS server (e.g. using GIS tools like GeoServer [23] or MapServer [24]) or WMTS (using tools like e.g. MapProxy [29])
  - Custom functionalities to export them (e.g. download from the Web Portal) as Shapefile or GeoPackage
  - Custom functionalities to export them to Google Maps / Google Earth [30] (widely used GIS applications)
- SensorML services and related standard software interfaces (see Section 2.1) to disseminate sensor measurements

3.2.3 Repository concept for enabling re-use

The machine learning and data mining community has a long history in sharing and re-using datasets to test and benchmark algorithms and develop new learning concepts. We give now a list of the most well-known such repositories:

- The oldest machine learning repository is the one maintained by the University of California, Irvine, currently hosting more than 300 datasets [6]. Datasets there usually come in the form of .names and .data files with the first providing the metadata describing the dataset and the second being a simple .csv file containing the actual data.
- A rather recent repository is mldata.org hosted by the machine learning group at the technical university of Berlin [27], which has been supported by the Pascale network [28]. It contains more than 800 datasets described in a number of different formats such as HDF5 (503.6 MB), XML, CSV, ARFF, LibSVM, Matlab, Octave
- KDD nuggets maintain a repository of data repositories [29].

The above repositories can be used for dissemination and re-use of the data generated by RAWFIE within the Machine Learning (ML) and Data Management (DM) communities.

Another option for sharing and re-use could be the exploitation of the Linked Open Data initiative and the reuse if appropriate of the technologies that have been developed in a number of European initiatives and projects in the area of open data such as: DaPaas [31], PlanetData [32]. However this might be less appropriate for the kind of data that will be generated by the RAWFIE infrastructure since Linked Open Data deal basically with the description of entities, their properties and their relations to other entities, while in RAWFIE we will be mainly generating measurements data.

4 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.

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
- 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 3.2.2). These repositories are free therefore there will not be 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.

References

[1] Sensor Observation Service (SOS): http://www.opengeospatial.org/standards/sos

[2] Sensor Model Language (SensorML): http://www.opengeospatial.org/standards/sensorml

[3] Observations and Measurements(O&M): http://www.opengeospatial.org/standards/om

[4] Sensor Planning Service: http://www.opengeospatial.org/standards/sps

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

[6] University of California Irvine machine learning repository: https://archive.ics.uci.edu/ml/datasets.html

[7] 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.

[8] Open Geospatial Consortium - OGC: http://www.opengeospatial.org/

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

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

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

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

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

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

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

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

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

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

[19] MathWorks : http://es.mathworks.com/help/robotics/examples/working-with-specialized-ros-messages.html

[20] Gazebo Simulator: http://gazebosim.org/

[21] ROS sensor format: http://wiki.ros.org/sensor_msgs

[22] ROS geometry format: http://wiki.ros.org/geometry_msgs

[23] GeoServer: http://geoserver.org/

[24] MapServer: http://mapserver.org/

[25] MapProxy: http://mapproxy.org/

[26] Google Earth: http://www.google.com/earth/

[27] Mldata.org, machine learning repository at the Technical University of Berlin: http://mldata.org/
[28] Pascal network of excellence: http://www.pascal-network.org
[29] KDD nuggets data repository: http://www.kdnuggets.com/datasets/index.html
[30] FP7 EU project: DaPaaS - A data-and-platform-as-a-service approach to efficient open data publication and consumption, 2007-2013, http://project.dapaas.eu/
[31] FP7 EU project: PlanetData, a European network of excellence on large scale data management.