SYnergy of integrated Sensors and Technologies for urban sEcured environMent

D1.2
DATA RETRIEVAL DEFINITION

25 June 2019 **V3.0**

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 787128

Project title	SYnergy of integrated Sensors and Technologies for urban sEcured environment
Project acronym	SYSTEM
Project number	787128
Start date of the project	1st September, 2018
Duration	36 months
Topic	SEC-10-FCT-2017. Integration of detection capabilities and data fusion with utility providers' network

Deliverable number	D1.2
Deliverable title	Data retrieval definition
Leading partner	RESI
Partners contributing	RESI, SCP, WUT, HSF, FHG, T4i
WP of reference	WP1
Title of the WP of reference	SCENARIO DEFINITION, USERS AND SYSTEM REQUIREMENTS
Task of reference	T1.1, T1.3, T1.5
Title of the task of reference	T1.1 REVISION OF TECHNOLOGY INVENTORY T1.3 REVISION OF DEFINED PROJECT SCENARIOS T1.5 SPECIFICATION ON SUBSYSTEM SOFTWARE AND DATA INTEGRATION
Deliverable type	Demonstrator
Dissemination level	PUBLIC **(with the request to maintain it temporarily CONFIDENTIAL only for members of the Consortium (including the Commission Services)**
Due date	M10 – June 2019. Foreseen in the DoA at M6 – February 2019. Extended by Ms. Oczko-Dolny (DG HOME) with the email sent to Ms. Cavallini (FORMIT) on 12 February 2019. Request of extension after the Project Management Board Meeting and the Executive Board held on 11-12 June in Bratislava (Slovakia). The request was done by email on 11 June 2019 by Ms. Cavallini (FORMIT) to Ms. Longo (REA).

Keywords	Interface, protocol, REST, CoAP, JSON, sensors, data retrieval
Abstract	This document defines the interfaces and formats to be used for controlling and retrieving measurements from each sensor (WATER/AIR) in order to collect them in the Monitoring Centre. The Monitoring Centre (MC) is a logically centralized server that gathers and collects field device data via a secured communication channel. It offers APIs to access and elaborate the stored data, according to a fully configurable set of privileges. Access to data undergoes a strict tracking policy and the Monitoring Centre assures that data cannot be altered nor accessed without leaving traces in the system. Furthermore, the Monitoring Centre ensures data integrity and non-repudiability.

	In this document the protocols and the interfaces between the monitoring centre and the sensors are described, in order to define how each system component interacts with the Monitoring Centre. The communication between the MC and each sensor is implemented either via REST web services (with payload in JSON) or CoAP (Constrained Application Protocol). The communication between each component and the MC is remote, wireless and asynchronous: the MC is always available remotely, no direct cable connection is allowed and data to be sent to the MC shall be transmitted via Internet. This document describes the data structure in according with the technical partners in the consortium.

Editor	Manuel Mastrofini (RESI), Pietro Guerrieri (RESI)
Contributors	Manuel Mastrofini (RESI), Pietro Guerrieri (RESI), Roberto Simmarano (SCP), Fernando Solano (BTEC), Jerzy Weremczuk (WUT), George C. Pallis (T4i)
Reviewers	Fernando Solano (BTEC), Roberto Mugavero (OSDIFE)
Submission date of the draft to reviewers	11 March 2019
Submission date of the draft to the SAB (if required)	Not required

Register of document versions

Partner acronym	Version number	Date	Suggested relevant changes	Notes
RESI	V1.0	11/03/2019	First draft	
WUT	V2.0	14/03/2019		
OSDIFE	V2.1	n.a	n.a.	
RESI	V2.2	26/03/2019	None	
FORMIT	V3.0	25/06/2019	None	

Every information is updated to the date of issue of this document

This document is composed by 23 pages

Table of Contents

Executive Summary 1
1 Main elements of this deliverable 2
1.1 Input from other projects 2
1.2 Input from other WPs and relation with other SYSTEM deliverables 2
1.3 Applicability 2
1.4 Reference documents 2
1.5 Purpose of the document 2
1.6 Structure of the document 2
2 Data Retrieval Definition 3
2.1 Software Interface Definition 3
2.1.1 REST and JSON 3
2.1.1.1 REST architectural style 3
2.1.1.2 JSON: Javascript Object Notation 4
2.2 CoaP 4
3 Data Retrieval Definition: Guidelines 5
4 Data Retrieval Definition for Water Sensors 6
4.1 Micromole pH Sensor (FhG-IZM) 6
4.2 Micromole Conductivity Sensor (FhG-IZM) 7
4.3 Smart Cable Water (Senshichips) 8
4.4 Smart Cable Water – pH Sensor (Senshichips) 9
4.5 Smart Cable Water – conductivity Sensor (Senshichips) 10
4.6 Millomole Sensors (CS) 11
4.7 Liquid Chromatography – Mass Spectrometry sensing subsystem (HSF) 12
4.8 Sampling & Storage Module (WUT) 13
5 Data Retrieval Definition for Gas Sensors 15
5.1 T4i Dover Gas Sensor (T4i) 15
5.2 Smart Cable Air Sensor (Sensichips) 16
6 Bibliography 18

List of Tables

Table 1 - MICROMOLE PH SENSOR 7
Table 2 - MICROMOLE CONDUCTIVITY SENSOR 8
Table 3 - SENSHICHIPS SMART CABLE WATER 9
Table 4 - SMART CABLE WATER – PH SENSOR 10
Table 5 - SMART CABLE WATER – CONDUCTIVITY SENSOR 11
Table 6 - MILLOMOLE SENSORS 12
Table 7 - LIQUID CHROMATOGRAPHY – MASS SPECTROMETRY SENSING SUBSYSTEM 13
Table 8 - SAMPLING & STORAGE MODULE 14
Table 9 - T4I DOVER GAS SENSOR 16
Table 10 - SENSICHIPS SMART CABLE AIR SENSOR 17

List of acronyms and abbreviations

CA	Consortium Agreement
CoAP	Constrained Application Protocol
COTS	Commercial Off The Shelf
DoA	Description of Action
EB	Executive Board
EC	European Commission *or* Electrical Conductivity
ES	Exploitation Strategy
GA	Grant Agreement
HTTP	HyperText Transfer Protocol
JSON	JavaScript Object Notation
LEA	Law Enforcing Agency
MC	Monitoring Centre
ReST	Representational State Transfer
SME	Small and Medium Enterprise

EXECUTIVE SUMMARY

This document aims at defining the interfaces and formats to be used for controlling and retrieving measurements from each sensor (WATER/AIR) in order to collect them in the Monitoring Centre (MC).

§2 Data Retrieval Definition includes the definition of the software interface control document that describes how each system component (sensors) interacts with the MC.

The communications between the MC and each sensor are specified in *§2.1 Software Interface Definition*. In particular, the protocols and the interfaces are defined in:

- *§ 2.1.1 REST and JSON*
- *§ 2.1.2 CoaP*

This document follows the classification of sensors provided in D1.1, according to the nature of the, i.e.:

- Water sensors and all subsystems (see *§4 Data Retrieval Definition for Water Sensors*)
- Air sensors (see *§5 Gas Sensors*)

This document establishes the technological aspects of the software integration between the Monitoring Centre (MC) and the sensors. In particular, protocols and formats will be better defined and finalized during the SYSTEM project life.
Specific details of the exchanged data may be affected by minor changes according to the emerging needs of the LEAs and, more in general, of the stakeholders.

1 MAIN ELEMENTS OF THIS DELIVERABLE

1.1 INPUT FROM OTHER PROJECTS

SYSTEM project builds mainly on top of the outcome of the NOSY project and of the MicroMole project. In particular, the deliverable *Data Retrieval Definition* (D1.2) describes the software integration between NOSY and Micromole sensing components and the Monitoring Centre (MC); therefore, the specific input from the two mentioned projects consists in the specifications of the interfaces of the sensors and their integration capabilities.

1.2 INPUT FROM OTHER WPS AND RELATION WITH OTHER SYSTEM DELIVERABLES

This document leverages on deliverable *Requirements, scenario definition and SYSTEM concept* (D1.1) and, in particular, on the technology inventory.

1.3 APPLICABILITY

The deliverable *Data Retrieval Definition* (D1.2) aims at becoming the reference point for the technological aspects of the software integration between the Monitoring Centre (MC) and the sensors. In particular, in terms of protocols and formats, taking into account the progress of the activities especially those foreseen in real environment.

1.4 REFERENCE DOCUMENTS

In order to set a framework in matter of a conflict between the Project Operational and Management Plan (D12.1) and other documents such as the Description of Actions (DoA) or the Grant Agreement, the following hierarchy will be applied:

1. Grant Agreement (GA);
2. Consortium Agreement (CA);
3. The Project Operational and Management Plan (D12.1).

The hierarchy related to the documents above implies that the latter document needs to be consistent with the former. In case of issues, this hierarchy of documents is mandatory.

1.5 PURPOSE OF THE DOCUMENT

This document defines the interfaces and formats to be used for controlling and retrieving measurements from each sensor (WATER/AIR) in order to collect them in the Monitoring Centre.

1.6 STRUCTURE OF THE DOCUMENT

The document is structured in the following way:

1. **Main elements of this deliverable;**
2. **Data Retrieval Definition** - interface and format to be used for controlling and data retrieval from each subsystem (Sensors) to the Monitoring Centre
3. **Data Retrieval Definition: Guidelines** - Guidelines and data definition related to how MC acquires metadata and measurements
4. **Data Retrieval Definition for Water Sensors** - Data and interface structure for Water sensors
5. **Data Retrieval Definition for Gas Sensors** - Data and interface structure for Gas sensors

2 Data Retrieval Definition

Task 1.5 the interface and format to be used for controlling and data retrieval from each subsystem to the data fusion and management station. According to this, each technology developer will extend the telecommunication capabilities of its devices to provide means for remote data fetching and basic device status. At the end of this task, each subsystem should define and possibly implement a design to be able to transmit to the central station.
The central station is the GENESI Monitoring Centre (MC), which is the logically centralised point of aggregation of sensors measurements where:

- data collected can be sent via Internet
- data received are securely stored and their integrity ensured
- data stored can be accessed, filtered, exported and visualized by authorized users
- data collected are correlated and correlation results are presented to the users

This document is a software interface control document that describes how each system component interacts with the MC, i.e. means, protocols and data exchanged.
The communication between each component and the MC is remote, wireless and asynchronous: the MC is always available remotely, no direct cable connection is allowed and data to be sent to the MC shall be transmitted via Internet.
According to the results of Task 1.1., "Revision of technology inventory", as reported in D1.1, an interface for data retrieval needs to be specified for each sensor.

2.1 Software Interface Definition

The communication between the MC and each sensor is implemented either via:

- REST web services, with payload in JSON
- Constrained Application Protocol (CoAP)

The first solution works on top of HTTP as transport protocol, so to ease the work on the sensor side. In fact, HTTP is a wide-spread protocol likely to be implemented in most embedded systems (either natively or via COTS). This way, no additional in-house software development effort is in charge of partners providing the sensors.
CoAP works very similarly to the REST and JSON solution with the main difference being that the headers are compressed and it mainly works on top of UDP (not TCP/HTTP).
In the next paragraphs an overview of the two solutions is provided.

2.1.1 REST and JSON

2.1.1.1 REST architectural style

REST (Representational State Transfer) is an architectural style for designing distributed systems. It is not a standard but a set of constraints, such as being stateless, having a client/server relationship, and a uniform interface. REST is not strictly related to HTTP, but it is most commonly associated with it.
The main principles of REST are:

- Resources expose easily understood directory structure URIs.
- Representations transfer JSON or XML to represent data objects and attributes.
- Messages use HTTP methods explicitly (for example, GET, POST, PUT, and DELETE).
- Stateless interactions store no client context on the server between requests. State dependencies limit and restrict scalability. The client holds session state.

In more technical terms: GET requests used to retrieve information. GET requests must be safe and idempotent, meaning regardless of how many times it repeats with the same parameters, the results are the same. They can have side effects, but the user doesn't expect them, so they cannot be critical to the operation of the system. Requests can also be partial or conditional. POST requests are used to create a new entity, but sometimes also to update an entity. PUT behave similarly to POST, with the addition they are idempotent and for this reason PUT requests are generally used for updates to the entity. PATCH is similar to PUT, but may apply updates to a portion of an entity. DELETE requests aim to remove an entity.

2.1.1.2 JSON: JAVASCRIPT OBJECT NOTATION

JSON is a language-independent data format (see http://www.json.org). It derives from JavaScript, but as of 2016, code to generate and parse JSON-format data is available in many programming languages and is widely used to encapsulate data exchanged by RESTful web services.
The JSON language format, originally defined by Douglas Crockford, today is defined by two competing standards:

- ECMA-404, describing only the allowed syntax;
- RFC 7159 providing also some semantic and security considerations.

Each JSON document format is defined by a JSON Schema (media type: application/schema+json; see for further details the Internet Engineering Task Force (IETF) Internet Draft https://tools.ietf.org/html/draft-zyp-json-schema-04), which specifies a JSON-based format to define the structure of JSON data for validation, documentation, and interaction control. A JSON Schema provides a contract for the JSON data required by a given application, and how that data can be modified. The JSON data schema can also be used to validate JSON data.
A JSON Schema contains, for example:

- the “$schema” directive
- the title of the schema
- a list of elements, based on primitives types:
  - Array, a JSON array,
  - Boolean, a JSON Boolean,
  - Integer, a JSON number without a fraction or exponent part,
  - Number, any JSON number, number includes integer;
  - String, a JSON string;
  - Null, the JSON null value;
  - Object, a JSON object, which is composed, in turns, by the above mentioned primitive types;
- the “required” directive, a JSON array of string with the list of required elements
- the reference to other imported schema

2.2 COAP

The Constrained Application Protocol (CoAP) is standardized in RFC 7252 from the IETF. It follows the same REST architecture and implements the basic four methods of HTTP: GET, POST, PUT and DELETE, but not the PATCH method.
Having in mind the reduced amount of memory and processor capabilities of embedded devices, CoAP was originally designed to work over User Datagram Protocol (UDP), as to avoid costly implementations of TCP in these resource-constrained devices.
Still, CoAP includes the following features of TCP to compensate for a lossy medium:

- Acknowledgement of reception for both requests and responses
- Retransmissions
- CoAP echo requests for diagnostics
- Token and Message id for identification of retransmissions and matching of request-responses

More recently, the IETF has standardized in RFC 7641 an extension of CoAP allowing for an efficient way of reporting measurements from constrained device. In this way, a CoAP client must only send one GET Request to a CoAP server with the observe flag set in order to subscribe to further changes on the value of the requested resource. The CoAP server will, when needed, reply and include in the response the value of the resource. Therefore, only one request is sent, but multiple response message can be received by the CoAP client, reducing the overall number of exchanged messages and the implied energy used on both sides.

3 DATA RETRIEVAL DEFINITION: GUIDELINES

When retrieving data from sensors, the MC acquires both metadata and measurements.

Metadata consist in:

- Device Id (mandatory): global unique identifier of the device, e.g. serial number. The serial number allows the MC to recognize and then authorize the entity that is transmitting data.
- Data Transmission Timestamp (mandatory): time when the transmission is happening (epoch time).
- Battery autonomy (mandatory): a decimal number indicating the amount of remaining battery
- Memory autonomy (mandatory): a decimal number indicating the amount of remaining storage memory.
- Health check status (mandatory)s: integer indicator to highlight when the sensor situation is critical. The number 0 indicates no criticality; 1 indicates the presence of a warning; 2 indicates the presence of a fault on the device.
- Health check message (optional): diagnostic message to provide better information concerning the health check.
- Transmission coordinates (optional): latitude, longitude and height of the transmission
- Configuration applied (optional): information about sensor-specific configuration, e.g. calibration information or threshold.
- Additional metadata (optional): key/value pairs for sharing additional information with the MC

Measurements are structured as follows:

- Measurement timestamp: time when the measurement has been performed (epoch time).
- Measurement type (optional): the type of detection executed (e.g. pH value, environment temperature, relative humidity).
- Measurement subtype (optional): if the sensor can perform multiple detections, the subtype of the current detection)
- Measurement coordinates (optional): latitude, longitude and height of the measurement.
- Measured value (mandatory): decimal number of the collected measure.
- Unit of measure (optional): unit of measure according to the International System of Units

4 DATA RETRIEVAL DEFINITION FOR WATER SENSORS

4.1 MICROMOLE PH SENSOR (FHG-IZM)

SENSOR DESCRIPTION TABLE	
Sensor name (reference)	pH sensor
Applicable to which utility network	sewage
Medium	water
Measuring principle	ISFET
DATA RETRIEVAL DEFINITION	
Communication type	Asynchronous messages
Communication style	CoAP
Payload format	JSON
Payload semantics	pH value

Payload example	{
	"deviceId": "123555AAA",
	"dataTransmissionTimestamp": 1551610635,
	"diagnostics":{
	"battery": 5.0,
	"storage": 44.6,
	"healthCheckStatus": “2”,
	"healthCheckMessage": “critical battery level”
	},
	"transmissionPosition": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"configuration": {
	“cycle”: “1.5”,
	“operationMode”: “M1”
	},
	"measurements":
	[
	{
	"timestamp": "1551618635",
	"type": “pH”,
	"position": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"value": 6.5,
	"unit": “adimensional”
	}, {
	"timestamp": "1551614444",
	"type": “pH”,
	"position": {
	"lat": 41.234371185302734,

	"lon": 14.815488815307617, "height": 25 }, "value": 6.2, "unit": "adimensional" }] }
Notes	

Table 1 - MICROMOLE PH SENSOR

4.2 MICROMOLE CONDUCTIVITY SENSOR (FHG-IZM)

SENSOR DESCRIPTION TABLE	
Sensor name (reference):	conductivity sensor
Applicable to which utility network:	sewage
Medium:	water
Measuring principle:	two electrode impedance measurement
DATA RETRIEVAL DEFINITION	
Communication type	Asynchronous messages
Communication style	CoAP
Payload format	JSON
Payload semantics	conductivity value

Payload example	{
	"deviceId": "123555BBB",
	"dataTransmissionTimestamp": 1551610635,
	"diagnostics":{
	"battery": 25.0,
	"storage": 44.6,
	"healthCheckStatus": "2",
	"healthCheckMessage": "hardware failure"
	},
	"transmissionPosition": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"configuration": {
	"cycle": "1.5",
	"operationMode": "M1"
	},
	"measurements":
	[
	{
	"timestamp": "1551618635",
	"type": "conductivity",
	"position": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,

	"height": 25
	},
	"value": 0.000052,
	"unit": "Siemens/meter"
	}, {
	"timestamp": "1551614444",
	"type": "conductivity",
	"position": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"value": 0.0000223,
	"unit": "Siemens/meter"
	}
	]
	}

Notes	

Table 2 - MICROMOLE CONDUCTIVITY SENSOR

4.3 SMART CABLE WATER (SENSHICHIPS)

SENSOR DESCRIPTION TABLE	
Sensor name (reference):	**Smart Cable Water** (Temperature, pH, ORP, Electrochemical, Impedance)
Applicable to which utility network:	sewage
Medium:	water
Measuring principle:	Oxidation/Reduction Potentials, Catalysis, Conductivity, Permittivity, EIS Spectroscopy
DATA RETRIEVAL DEFINITION	
Communication type	Asynchronous messages
Communication style	REST
Payload format	*JSON*
Payload semantics	Analyte List Classification

Payload example	{
	"deviceId": "123555BBB",
	"dataTransmissionTimestamp": 1551610635,
	"diagnostics":{
	"battery": 25.0,
	"storage": 44.6,
	"healthCheckStatus": "2",
	"healthCheckMessage": "hardware failure"
	},
	"transmissionPosition": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"configuration": {

	“cycle”: "1.5",
	“operationMode”: “M1”
	},
	
	"analytesClassification":
	[
	{
	“analyte”: "AMMONIA",
	“probability”: "89%",
	“concentration”: "10 ppm"
	},
	{
	“analyte”: "ACETONE",
	“probability”: "10%",
	“concentration”: "2 ppm"
	},
	{
	“analyte”: "ALCOHOL",
	“probability”: "1%",
	“concentration”: "1 ppm"
	}, others analytes..
	]
	}

Notes	

Table 3 - SENSHICHIPS SMART CABLE WATER

4.4 SMART CABLE WATER – pH SENSOR (SENSHICHIPS)

SENSOR DESCRIPTION TABLE	
Sensor name (reference):	**Smart Cable Water - pH**
Applicable to which utility network:	sewage
Medium:	water
Measuring principle:	Oxidation/Reduction Potentials, Catalysis, Conductivity, Permittivity, EIS Spectroscopy
DATA RETRIEVAL DEFINITION	
Communication type	Asynchronous messages
Communication style	REST
Payload format	*JSON*
Payload semantics	pH

Payload example	{
	"deviceId": "123555AAA",
	"dataTransmissionTimestamp": 1551610635,
	"diagnostics":{
	"battery": 5.0,
	"storage": 44.6,
	"healthCheckStatus": “2”,
	"healthCheckMessage": “critical battery level”
	},
	"transmissionPosition": {
	"lat": 41.234371185302734,

	"lon": 14.815488815307617,
	"height": 25
	},
	"configuration": {
	"cycle": "1.5",
	"operationMode": "M1"
	},
	"measurements":
	[
	{
	"timestamp": "1551618635",
	"type": "pH",
	"position": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"value": 6.5,
	"unit": "adimensional"
	}
	]
	}

Notes	

Table 4 - SMART CABLE WATER – PH SENSOR

4.5 Smart Cable Water – conductivity Sensor (Senshichips)

SENSOR DESCRIPTION TABLE	
Sensor name (reference):	**Smart Cable Water - conductivity**
Applicable to which utility network:	sewage
Medium:	water
Measuring principle:	Conductivity
DATA RETRIEVAL DEFINITION	
Communication type	Asynchronous messages
Communication style	REST
Payload format	*JSON*
Payload semantics	conductivity

Payload example	{
	"deviceId": "123555BBB",
	"dataTransmissionTimestamp": 1551610635,
	"diagnostics":{
	"battery": 25.0,
	"storage": 44.6,
	"healthCheckStatus": "2",
	"healthCheckMessage": "hardware failure"
	},
	"transmissionPosition": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25

	},
	"configuration": {
	"cycle": "1.5",
	"operationMode": "M1"
	},
	"measurements":
	[
	{
	"timestamp": "1551618635",
	"type": "conductivity",
	"position": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"value": 0.000052,
	"unit": "Siemens/meter"
	}
	]
	}

Notes	

Table 5 - SMART CABLE WATER – CONDUCTIVITY SENSOR

4.6 MILLOMOLE SENSORS (CS)

SENSOR DESCRIPTION TABLE	
Sensor name (reference):	milliMole
Applicable to which utility network:	Sewage water, manhole
Medium:	Water
Measuring principle:	Competitive immune/amperometric detection
DATA RETRIEVAL DEFINITION	
Communication type	Asynchronous messages
Communication style	CoAP
Payload format	JSON
Payload semantics	concentration value

Payload example	{
	"deviceId": "123555BBB",
	"dataTransmissionTimestamp": 1551610635,
	"diagnostics":{
	"battery": 25.0,
	"storage": 44.6,
	"healthCheckStatus": "2",
	"healthCheckMessage": "hardware failure"
	},
	"transmissionPosition": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},

	"configuration": {
	"cycle": "1.5",
	"operationMode": "M1"
	},
	"measurements":
	[
	{
	"timestamp": "1551618635",
	"type": "concentration",
	"subtype": "ether",
	"position": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"value": 30,
	"unit": "ppm"
	}, {
	"timestamp": "1551618635",
	"type": "concentration",
	"subtype": "ethanol",
	"position": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 44
	},
	"value": 30,
	"unit": "adimensional"
	}
	]
	}

Notes	

Table 6 - MILLOMOLE SENSORS

4.7 Liquid Chromatography – Mass Spectrometry sensing subsystem (HSF)

SENSOR DESCRIPTION TABLE	
Sensor name (reference):	Liquid Chromatography (LC) – Mass Spectrometry (MS) system
Applicable to which utility network:	Sewage
Medium:	Water
Measuring principle:	Measuring the *m/z* of target substances
DATA RETRIEVAL DEFINITION	
Communication type	Asynchronous messages
Communication style	REST APIs
Payload format	JSON
Payload semantics	mass-to-charge value
Payload example	{ "deviceId": "123555BBB", "dataTransmissionTimestamp": 1551610635,

	"diagnostics":{
	"battery": 25.0,
	"storage": 44.6,
	"healthCheckStatus": "2",
	"healthCheckMessage": "hardware failure"
	},
	"transmissionPosition": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"configuration": {
	"cycle": "1.5",
	"operationMode": "M1"
	},
	"measurements":
	[
	{
	"timestamp": "1551618635",
	"type": "mass-to-charge",
	"position": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"value": 1000,
	"unit": "kg/Coulomb"
	}
	]
	}

Notes	

Table 7 - LIQUID CHROMATOGRAPHY – MASS SPECTROMETRY SENSING SUBSYSTEM

4.8 SAMPLING & STORAGE MODULE (WUT)

SENSOR DESCRIPTION TABLE	
Sensor name (reference):	Sampling and Storage Module (S2M)
Applicable to which utility network:	Sewage
Medium:	Water
Measuring principle:	Gathering liquid samples or using solid state absorbents
DATA RETRIEVAL DEFINITION	
Communication type	Asynchronous messages
Communication style	REST APIs
Payload format	JSON
Payload semantics	Number of samples gathered, number of free compartments, commands: change the sample acquisition time , catch the sample immediately
Payload example	{ "deviceId": "123555BBB", "dataTransmissionTimestamp": 1551610635,

	"diagnostics":{
	"battery": 25.0,
	"storage": 44.0,
	"healthCheckStatus": "2",
	"healthCheckMessage": "hardware failure"
	},
	"measurements":
	[
	{
	"timestamp": "1551618635",
	"type": "storeUsage",
	"value": 10,
	"unit": "adimensional"
	}
	]
	}

Notes	

Table 8 - SAMPLING & STORAGE MODULE

5 DATA RETRIEVAL DEFINITION FOR GAS SENSORS

5.1 T4I DOVER GAS SENSOR (T4I)

SENSOR DESCRIPTION TABLE	
Sensor name (reference):	T4i DOVER™
Applicable to which utility network:	Air (open space) or waste headspace (confined space)
Medium:	Air
Measuring principle:	GC-PID
DATA RETRIEVAL DEFINITION	
Communication type	Asynchronous messages
Communication style	REST APIs
Payload format	JSON
Payload semantics	concentration value

Payload example	{
	"deviceId": "123555BBB",
	"dataTransmissionTimestamp": 1551610635,
	"diagnostics":{
	"battery": 25.0,
	"storage": 44.6,
	"healthCheckStatus": “2”,
	"healthCheckMessage": “hardware failure”
	},
	"transmissionPosition": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"configuration": {
	“cycle”: “1.5”,
	“operationMode”: “M1”
	},
	"measurements":
	[
	{
	"timestamp": "1551618635",
	"type": “concentration”,
	"subtype": “ether”,
	"position": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"value": 30,
	"unit": “ppm”
	}, {
	"timestamp": "1551618635",
	"type": “concentration”,
	"subtype": “ethanol”,
	"position": {

	"lat": 41.234371185302734, "lon": 14.815488815307617, "height": 44 }, "value": 30, "unit": “ppm” }] }
Notes	

Table 9 - T4I DOVER GAS SENSOR

5.2 Smart Cable Air Sensor (Sensichips)

SENSOR DESCRIPTION TABLE	
Sensor name (reference):	**Smart Cable Air** (Temperature, Aluminium Oxide for Ammonia, Polyamide for Vapours, Graphene for Oxidants, Commercial High Temp MOX for VOCs)
Applicable to which utility network:	sewage, waste trucks and air
Medium:	gas
Measuring principle:	Chemisorption, RedOx
DATA RETRIEVAL DEFINITION	
Communication type	Asynchronous messages
Communication style	REST
Payload format	*JSON*
Payload semantics	environment temperature, ph, OR potential, impedance

Payload example	{
	"deviceId": "123555BBB",
	"dataTransmissionTimestamp": 1551610635,
	"diagnostics":{
	"battery": 25.0,
	"storage": 44.6,
	"healthCheckStatus": “2”,
	"healthCheckMessage": “hardware failure”
	},
	"transmissionPosition": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"configuration": {
	“cycle”: “1.5”,
	“operationMode”: “M1”
	},
	"measurements":
	[
	{
	"timestamp": "1551618635",
	"type": “temperature”,
	"position": {

	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"value": 15.4,
	"unit": “Celsius”
	}, {
	"timestamp": "1551614444",
	"type": “pH”,
	"position": {
	"lat": 41.234371185302734,
	"lon": 14.815488815307617,
	"height": 25
	},
	"value": 0.0000223,
	"unit": “Siemens/meter”
	}
	]
	}

Notes	

Table 10 - SENSICHIPS SMART CABLE AIR SENSOR

6 BIBLIOGRAPHY

No specific references have been used.