



Deliverable D4.7

Framework and standards for data interoperability - version 2

Work Package 4

Pilots, data and campaigns

Version: Final



This project has received funding from the European Union's Horizon Europe Framework Programme under grant agreement N° 101057497.

Deliverable Overview

This deliverable represents the second version of the EDIAQI “Framework and standards for data interoperability” and provides a list of standards and technical specifications to implement a common interoperability approach and related technical solutions. The aim is to guarantee that EDIAQI data collected from pilots and campaigns will be shared and made accessible through adequate standards for interoperability.

Additional Information

Type: R – Document, report

Dissemination Level: PU – Public

Official Submission Date: 30th of November 2024

Actual Submission Date: 28th of November 2024

Disclaimer

This document contains material, which is the copyright of certain EDIAQI partners, and may not be reproduced or copied without permission. All EDIAQI consortium partners have agreed to the full publication of this document if not declared “Confidential”. The commercial use of any information contained in this document may require a license from the proprietor of that information. The reproduction of this document or of parts of it requires an agreement with the proprietor of that information.



Document Revision History

Version	Date	Description	Partners
V1.0 (Final D4.3)	15 th of December 2023	Final version D4.3 submitted	LC
V1.2	14 th of October 2024	New observed properties added in Table 5 Added new table (8.1). Updated section 5.2 Examples of 'insert' operations	DEDA
V1.3	25 th of October 2024	Added epidemiologic data glossary in new section Error! Reference source not found.	LC, TROPOS
V1.4	28 th of October 2024	Updated chapters 6 and 7	DEDA
V1.5	12 th of November 2024	Updated chapter 3	LC
V1.6	13 th of November 2024	Added table in chapter 7	DEDA
V1.7	18 th of November 2024	Internal Review	TROPOS
Final	28 th of November 2024	Final quality check	LC

Authors and Reviewers

Authors

- Piergiorgio Cipriano (DEDA)
- Giulia Degli Esposti (DEDA)
- Martina Forconi (DEDA)
- Chiara Savoldi (DEDA)
- Mario Lovric (LC)
- Ivan Notardonato (UNIMOL)
- Alessandro Battaglia (LAS)

Reviewers

- Tomislav Cernava (TU Graz)
- Honey Alas (TROPOS)



This project has received funding from the European Union's Horizon Europe Framework Programme under grant agreement N° 101057497.

Statement of Originality

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.



This project has received funding from the European Union's Horizon Europe Framework Programme under grant agreement N° 101057497.

Table of Contents

1. Executive Summary	12
2. Introduction	14
2.1 Rationale	14
2.2 Deliverable objectives	14
3. Semantic interoperability	16
3.1 Data parameters	16
3.2 Air Quality Data	18
3.3 Auxiliary Data	26
3.3 Epidemiologic Data	30
3.4 Biological Data	33
4. Technical interoperability.....	35
4.1 Standards.....	35
4.1.1 SensorThings API.....	37
4.1.2 Web Map Service interface (ISO19128)	40
4.1.3 Web Feature Service interface (ISO19142)	43
4.1.4 CityGML.....	45
4.2 Reference open-source software solutions.....	47
4.2.1 FROST (FRaunhofer Opensource SensorThings-Server)	47
4.2.2 PostgreSQL / PostGIS	48
4.2.3 Spatial server implementing WMS and WFS standard	48
4.2.4 QGIS and plugin SensorThings API	49
4.2.5 3DCityDB + QGIS plugin 3DCityDB Tool	50
5. Mapping EDIAQI concepts towards standard data models.....	51
5.1 STA data model	51
5.1.1 Things.....	54
5.1.2 Locations	56
5.1.3 FeatureOfInterest	60
5.1.4 Sensors.....	66



5.1.5	ObservedProperties	69
5.1.6	Observations	70
5.1.7	DataStream.....	71
5.2	Examples of ‘insert’ operations	73
5.2.1	Step 1. Create a Thing entity.....	74
5.2.2	Step 2. Create a Location entity.....	74
5.2.3	Step 3. Create a Sensor entity.....	75
5.2.4	Step 4. Create an Observed Property entity	75
5.2.5	Step 5. Create a FeatureOfInterest entity.....	76
5.2.6	Step 6. Create a Datastream entity.....	77
5.2.7	Step 7. Create an Observation entity.....	78
5.3	Other standard data models to be considered	79
5.3.1	CityGML “Building” class.....	79
5.3.2	INSPIRE Directive “Building” data theme.....	80
6.	Status of deployments.....	82
6.1	Indoor air quality and auxiliary data.....	82
6.1	Outdoor air quality data.....	88
7.	Annexes	89
7.1	Mapping towards CityGML data model	89
7.2	Standards for metadata (FAIR) and solutions for discovery capabilities.....	91



List of Figures

Figure 1 - Entities (classes) of the SensorThings data model.....	38
Figure 2 - Level Of Details (LOD) of CityGML	46
Figure 3 - Radiello passive sampler	55
Figure 4 - NetPID monitoring unit	55
Figure 5 - Location of the Thing corresponding to building address (see it on GoogleMaps) 60	
Figure 6 - Exact location of the Thing (see it on GoogleMaps)	60
Figure 7 - Example of FeatureOfInterest corresponding to a single room	65
Figure 8 - Sensorion SPS30 for PM2.5	68
Figure 9 - MH-410D NDIR Infrared CO2.....	68
Figure 10 - Simplified UML Class Diagram for “Building” in CityGML.....	79
Figure 11 - Overview of the Building Base in INSPIRE Technical Guidelines (TG).....	80
Figure 12 - CityGML LoD2, LoD3 and LoD4 in INSPIRE “Buildings” TG.....	81
Figure 13 - Database schema for “Building” in 3DCityDB.....	89
Figure 14 - High school "Aleotti" (Ferrara, Italy) indoor map at first floor	90
Figure 15 - FAIR data and EC requirements.....	91
Figure 16 - Standards for metadata in different domains	93



List of Tables

Table 1 Indoor air quality parameters measured with sensorics	18
Table 2 Environmental conditions recorded by sensorics	19
Table 3 Outdoor air quality parameters measured by sensorics.....	20
Table 4 Additional indoor air quality parameters.....	21
Table 5 Chemical components of indoor air.....	22
Table 6 Auxiliary environmental conditions	26
Table 7 Auxiliary occupancy parameters.....	27
Table 8 Auxiliary building parameters.....	28
Table 9 Auxiliary parameters (behaviours).....	29
Table 10 Epidemiological questionnaire for large scale building studies	30
Table 11 Toxicological parameters	32
Table 12 Lifestyle parameters collected.....	34
Table 13 Technical details for sensor data streaming services - client.....	39
Table 14 Technical details for sensor data streaming services - server.....	40
Table 15 Technical details for web map services - client.....	41
Table 16 Technical details for Map Services - server.....	42
Table 17 Technical details for Map Services - interface	43
Table 18 Technical details for Download Services - client.....	44
Table 19 Technical details for Download Services – server	45
Table 20 Technical details for Download Services - interface	45
Table 21 Mapping EDIAQI data towards STA data model: “Things” entity	54
Table 22 Mapping EDIAQI data towards STA data model: “Locations” entity	56
Table 23 Mapping EDIAQI data towards STA data model: “FeatureOfInterest” entity	61
Table 24 Mapping EDIAQI data towards STA data model: “Sensors” entity	66
Table 25 Mapping EDIAQI data towards STA data model: “ObservedProperties” entity.....	69
Table 26 Mapping EDIAQI data towards STA data model: “Observations” entity	70
Table 27 Mapping EDIAQI data towards STA data model: “DataStreams” entity	71
Table 28 Sensor Things API - Guidelines available related to integrity constraints.....	73
Table 29 General timeline for EDIAQI deployments.....	82



Table 30 Detailed plan for semantic interoperability.....	83
Table 31 Types of IAQ monitoring devices (with numbers and parameters)	85



List of Terms and Abbreviations

Abbreviation	Description
6LowPAN	IPv6 over Low-Power Wireless Personal Area Networks
ANT	Institute of Anthropological Research
API	Application Protocol Interface
AQ	Air Quality
CAS	Chemical Abstracts Service
CoAP	Constrained Application Protocol
CEN	Comité Européen de Normalisation
CQL	Common Query Language
CRS	Coordinate Reference System
CSV	Comma Separated Values
EC	European Commission
ECHA	European Chemical Agency
EIONET	European Environment Information and Observation Network
FROST	FRaunhofer Opensource SensorThings
FTP	File Transfer Protocol
GEMET	GEneral Multilingual Environmental Thesaurus
GML	Geography Markup Language
HTTP(S)	Hypertext Transfer Protocol (Secure)
INSPIRE	Infrastructure for Spatial Information in Europe
IAQ	Indoor Air Quality
IOT	Internet Of Things
IPChem	Information Platform for Chemical Monitoring
ISO	International Organization for Standardization
JPEG	Joint Photographic Experts Group
JSON	JavaScript Object Notation
KML	Keyhole Markup Language
KVP	Key Value Pair



LOD	Level Of Details
MQTT	MQ Telemetry Transport
OGC	Open Geospatial Consortium
OWS	OGC Web Services
PDF	Portable
PNG	Portable Document Format
REST	Representational state transfer
RDBMS	Relational Database Management System
SensorML	Sensor Markup Language
SLD	Styled Layer Descriptor
SQL	Structured Query Language
SSN	Semantic Sensor Network
STA	SensorThings API
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
WGS84	World Geodetic System 1984
WMS	Web Map Service
WFS	Web Feature Service
XML	eXtensible Markup Language



1. Executive Summary

This Deliverable defines the common interoperability approach in EDIAQI and reports on the implementation progress thus far. The interoperability is an important aspect in EDIAQI ensuring that data can be shared within the project and with third parties, whenever needed and in consensus with the deliverable D2.3 Data Management Plan – version 1.

This document is based on, **updates and replaces** the initial version of this deliverable (v1, D4.3) submitted in December 2023.

Compared to D4.3, this final version provides the updated list of parameters considered in EDIAQI for indoor air quality (IAQ), outdoor air quality, auxiliary data about the context, as well as epidemiologic and biological data. Moreover, it provides the list of the endpoints (servers) that have been installed to share IAQ data collected in pilot areas and campaigns. Indeed, in EDIAQI the interoperability is ensured by using standards for data sharing, representation and interfaces as well as including open-source components into the developed software solutions for EDIAQI. The data exchange needs adequate standards for interoperability, so that the involved parties can collect, transmit and share the data based on a common technical approach. It is necessary to establish common semantics described by machine-readable vocabularies ensuring the interpretability of the data throughout the data exchange and the foreseen analytics in EDIAQI. These semantics need to cover the elements of interest for indoor air quality studies, namely, the investigated buildings, the involved people and their respective risk level, real-time measurements of the air quality parameters and the subjective perception and complaints of people (chapter 3).

Taking well-established technical standards like the Indoor Air Quality System standard (ISO 16000) into account, a data organisation scheme was designed to ensure semantic interoperability. This schema consists of three main parts:

1. Metadata¹ providing information about the data.

¹ Standards for discovery metadata will be part of the next version of this deliverable (D4.7) in compliance to EU Directives and ISO technical specs (see also chapter **Error! Reference source not found.** with initial references to Dublin Core / ISO15836 and DCAT-AP).



2. Data, which is further divided into static and dynamic data, depending on the expected temporal resolution.
3. Auxiliary data describing the rooms where the data has been collected. Additionally, there are naming conventions to uniquely identify each data element. All in all, this is the necessary framework to describe what is measured in each pilot and give the technical details like units, sampling, and delivery rates.

The technical interoperability is ensured by complying to standards and by using reference software solutions already implementing these standards (chapter 0). When looking at one of the main standards considered, the OGC SensorThings API² is employed as a REST communication method to transmit measurement data in JSON format over HTTP(S). The SensorThings API handles the semantics to identify the measurement as well as the environment of the measurements that produced the data. Alongside the tabular data and diagrams for statistical data, visualisation services are employed to create maps and visual representations from spatialised data. These standards are implemented by the selected open-source software implementation, like the FROST (Fraunhofer Opensource SensorThings-Server) for implementing the OGC SensorThings API Part1: Sensing 1.0 (with PostgreSQL / PostGIS for storing measurements and geospatial data) and geographical servers, like Geoserver, for implementing ISO19128 (WMS) and ISO19128 (WFS) standards. One of the main steps will be to “map” EDIAQI concepts and semantics towards standard data models. Chapter 5 provides the initial version of a full mapping towards the entities defined by SensorThings.

The deliverable closes with a progress report on the activities in Task 4.3 so far and provides a general update about FROST server instances deployed by partners.

² <https://www.ogc.org/publications/standard/sensorthings/>



2. Introduction

2.1 Rationale

The activity undertaken in Task T4.3 contributes to the “data management plan and data standard interoperability”. Indeed, EDIAQI data fall within geographic information as they have a direct reference to the location where observation and measurements are taken; therefore, EDIAQI data will be transferred and handled through common standard formats and protocols already defined within the ISO 19100 series (starting from ISO 19156:2011 “Observation & Measurements”³).

Indoor Air Quality (IAQ) monitoring data will be collected according to a common interoperable technical approach, with common semantics described in machine-readable vocabularies. In task T4.3, a common methodology has been identified to be used for identifying and assessing IAQ elements and ensuring a continuous improvement of air hygiene and well-being across pilots.

The methodology followed also considers and integrates the beyond state-of-the-art technical standards on IAQ Management Systems (ISO 16000). The IAQ behavioural change monitoring campaigns will be driven by a process management which shall rest on the following pillars:

1. data on building and people for a risk level and assessment of IAQ aspects of? building categories.
2. real-time analytical measurements on IAQ via wireless remote multi-sensor devices.
3. collection of subjective perceptions and complaints about IAQ from residents.

2.2 Deliverable objectives

The goal of this document is to define and implement a common interoperability approach and related technical solutions to guarantee that EDIAQI data collected from pilots will be shared and made accessible through adequate standards for interoperability.

³ Revised by ISO 19156:2023.



The work presented here is based on an analysis of open standards defined by international organisations like the World Wide Web Consortium (W3C), the International Organisation for Standardisation (ISO), the European Committee for Standardization (CEN) and the Open Geospatial Consortium (OGC).

Particular attention has been dedicated to the availability of “open standards”, defined by OGC as “agreed specification of rules and guidelines about how to implement software interfaces and data encodings” and “freely and publicly available, non-discriminatory, with no license fees, vendor neutral, data neutral and based on Consensus”⁴.

Open standards defined at international level guarantee the implementation of software solutions (in particular, open-source solutions) that implement them and facilitate the interoperability of data and web-based services.

⁴ <https://opengeospatial.github.io/e-learning/ogc-standards/text/services-ogc.html>



3. Semantic interoperability

This chapter contains the semantics agreed upon within EDIAQI to achieve interoperability between the different pilots and campaigns. The data generated within EDIAQI can be categorised into the following:

- Metadata – data providing information about the data gathered from the pilots and campaigns following the IPChem template⁵.
- Data – measured air quality parameters
 - Dynamic data which are continuously updated by online measurements/access (data from sensorics, data accessed from external agencies, such as outdoor air pollution and meteorological data).
 - Static data which do not change frequently (particles collected on a filter in 1 indoor space over a specified amount of time without further repetition).
- Auxiliary data – similar to static data, but more constant (indoor parameters such as average occupancy, placement of sensors, number of windows and doors, building types etc.).

3.1 Data parameters

In this version of the deliverable, we have included the following parameters:

- indoor biological parameters.
- indoor chemical parameters.
- lifestyle metadata from questionnaires, and
- epidemiological data from questionnaires.

This final version also includes information about outdoor air pollution data collected and provided by the European Environmental Agency (chapter 6).

The semantics within the EDIAQI project are presented in the following tables. The meaning of each of the columns is described below:

- Parameter name – full name of the parameter measured/identified.

⁵ [IPChem Metadata Template](#)



- Common (C) vs pilot-specific – describes if the parameter is measured/identified across all pilots and campaigns or only in some:
 - C – common across all pilots and campaigns
 - P1 – Ferrara pilot
 - P2 – Tallinn pilot
 - P3 – Zagreb pilot
 - P4 – Filtration pilot
 - C2 – Seville campaign
 - C3 – Vilnius campaign
- UoM – unit of measure
- Symbol – parameter symbol
- Sampling resolution – how often measurements are taken
- Reporting resolution – how often are data shared to the project data platform
- Reporting values – which values are shared (e.g. min, max, average, standard deviation, ...) for a specific temporal unit (e.g. 10', 15', 20', 30', 60')

Initial vocabularies have been already identified and referenced in this chapter and in chapter 4):

- Air Quality Pollutants (source: EEA EIONET)⁶
- Air Quality Measurement Methods (source: EEA EIONET)⁷
- Air Quality Measurement Type (source: EEA EIONET)⁸
- Eurostat Units of Measures (source: EEA EIONET)⁹
- ECHA Chemicals database (source: ECHA)¹⁰
- CurrentUse of Buildings (source: GEMET INSPIRE Registry)¹¹

⁶ <https://dd.eionet.europa.eu/vocabulary/aq/pollutant/>

⁷ <http://dd.eionet.europa.eu/vocabulary/aq/measurementmethod/>

⁸ <http://dd.eionet.europa.eu/vocabulary/aq/measurementtype>

⁹ <https://dd.eionet.europa.eu/vocabulary/eurostat/unit/>

¹⁰ <https://echa.europa.eu/en/search-for-chemicals>

¹¹ <https://inspire.ec.europa.eu/codelist/CurrentUseValue>



3.2 Air Quality Data

This section contains a list of 5 tables, each one identifying the main characteristics of data foreseen in the different EDIAQI pilots and campaigns. For each parameter listed in the tables, details have been collected from EDIAQI partners to allow a synoptic view: name of the parameter, code and name of EDIAQI pilot area where the parameter is considered, the unit of measure suggested, its sampling resolution (in '' seconds or ' minutes), the value(s) to be shared with the EDIAQI data platform and the temporal unit (in ' minutes) the value is referred to.

Table 1 provides a list of air quality parameters measured in indoor locations (pilot buildings). To define common semantics, the name of each parameter is based on the official EIONET Vocabulary "Pollutant"¹² and it is linked to the exact URI containing its definition (URI is the URL to the EIONET concept, to be referenced in the data shared - see chapter 5, ObservedProperties). In EDIAQI, each parameter will be uniquely identified with a numeric ID.

Table 1 Indoor air quality parameters measured with sensorics

Parameter Name	ID	Common (C) vs PilotSpecific	UoM	Symbol	Sampling resolution	Values and temp. unit
Particulate matter < 10 µm (aerosol)	1	P1 – Ferrara	µg/m ³ (microgram per cubic meter)	PM10	e.g. P1: 5''	Average 5'
Particulate matter < 2.5 µm (aerosol)	2	C	µg/m ³ (microgram per cubic meter)	PM2.5	e.g. P1: 5''	Average 5'
Particulate matter < 1 µm (aerosol)	3	P2 – Tallinn C3 – Vilnius	µg/m ³ (microgram per cubic meter)	PM1	e.g. P1: 5''	Average 5'

¹² <http://dd.eionet.europa.eu/vocabulary/aq/pollutant/>



Carbon monoxide (air)	4	P1 – Ferrara P3 – Zagreb P4 – Filtration C3 – Vilnius	$\mu\text{g}/\text{m}^3$ (microgram per cubic meter)	CO	e.g. P1: 5''	Average 5'
Carbon dioxide (air)	5	C	ppm (parts per million)	CO ₂	e.g. P1: 5''	Average 5'
Ozone	6	P2 – Tallinn C3 – Vilnius	$\mu\text{g}/\text{m}^3$ (microgram per cubic meter)	O ₃	e.g. P1: 5''	Average 5'
Nitrogen dioxide (air)	7	P2 – Tallinn C3 – Vilnius	$\mu\text{g}/\text{m}^3$ (microgram per cubic meter)	NO ₂	e.g. P1: 5''	Average 5'
Total volatile organic compounds (air)	8	C	$\mu\text{g}/\text{m}^3$ (microgram per cubic meter)	TVOC	e.g. P1: 5''	Average 5'
Hydrocarbons, C9, aromatics	9	P1 – Ferrara	$\mu\text{g}/\text{m}^3$ (microgram per cubic meter)	AVOC	e.g. P1: 5''	Average 5'

Table 2 provides the list of parameters related to indoor environment to measure the thermal/climate conditions. In this table, the name of each parameter is also based on the EIONAT Vocabulary “Meteo parameter”¹³ and it is linked to the exact URI containing its definition.

Table 2 Environmental conditions recorded by sensorics

¹³ <http://dd.eionet.europa.eu/vocabulary/aq/meteoparameter/>



Parameter Name	ID	Common vs Pilot-specific	UoM	Indoor or outdoor or both	Symbol	Sampling resolution	Values and temp. unit
Temperature	10	C	°C (degrees Celsius)	both	T	e.g. P1: 5''	Average 5'
Relative humidity	11	C	% (percent)	both	RH	e.g. P1: 5''	Average 5'
Pressure	12	P1 – Ferrara P3 – Zagreb P4 – Filtration C3 – Vilnius	hPa (Hectopascal)	both	P	e.g. P1: 5''	Average 5'
Noise	13	P3 – Zagreb P4 – Filtration C3 – Vilnius	dB (decibels)	outdoor	noise	n.a.	Average 5'

Table 3 considers air quality parameters for outdoor conditions related to pilot buildings.

Table 3 Outdoor air quality parameters measured by sensorics

Parameter Name	ID	Common (C) vs PilotSpecific	UoM	Symbol	Sampling resolution	Value and temp.unit
Particulate matter < 10 µm (aerosol)	1	P1 – Ferrara P3 – Zagreb P4 – Filtration C3 – Vilnius	µg/m ³ (microgram per cubic meter)	PM10	e.g. P1: 5''	Average 5'
Particulate matter < 2.5 µm (aerosol)	2	C	µg/m ³ (microgram per cubic meter)	PM2.5	e.g. P1: 5''	Average 5'
Particulate matter < 1 µm (aerosol)	3	P2 – Tallinn P3 – Zagreb P4 – Filtration C3 – Vilnius	µg/m ³ (microgram per cubic meter)	PM1	e.g. P1: 5''	Average 5'



Carbon monoxide (air)	4	P3 – Zagreb P4 – Filtration C3 – Vilnius	$\mu\text{g}/\text{m}^3$ (microgram per cubic meter)	CO	e.g. P1: 5''	Average 5'
Carbon dioxide (air)	5	P2 – Tallinn C3 – Vilnius	ppm (parts per million)	CO ₂	e.g. P1: 5''	Average 5'
Ozone	6	P1 – Ferrara P2 – Tallinn P3 – Zagreb P4 – Filtration C3 – Vilnius	$\mu\text{g}/\text{m}^3$ (microgram per cubic meter)	O ₃	e.g. P1: 5''	Average 5'
Nitrogen dioxide (air)	7	P1 – Ferrara P2 – Tallinn P3 – Zagreb P4 – Filtration C3 – Vilnius	$\mu\text{g}/\text{m}^3$ (microgram per cubic meter)	NO ₂	e.g. P1: 5''	Average 5'
Total volatile organic compounds (air)	8	P1 – Ferrara P2 – Tallinn C3 - Vilnius	$\mu\text{g}/\text{m}^3$ (microgram per cubic meter)	TVOC	e.g. P1: 5''	Average 5'
Nitrogen monoxide (air)	14	P3 – Zagreb P4 – Filtration C3 – Vilnius	$\mu\text{g}/\text{m}^3$ (microgram per cubic meter)	NO	e.g. P1: 5''	Average 5'

Table 4 considers additional parameters measured in specific pilots and campaigns of EDIAQI project.

Table 4 Additional indoor air quality parameters

Parameter Name	ID	Common vs Pilot-specific	UoM	Symbol	Sampling resolution	Data sharing frequency
Equivalent black carbon	15	C3 (short campaign) P4 - Filtration	$\mu\text{g}/\text{m}^3$ (microgram per cubic meter)	eBC	1 minute	Once



This project has received funding from the European Union's Horizon Europe Framework Programme under grant agreement N° 101057497.

Aerosol particle number concentration	16	C3 (short campaign) P4 - Filtration	#/cm ³ (number of particles per cubic centimeter)	N	5 minutes	Once
Aerosol particle number size distribution	17	C3 (short campaign) P4 - Filtration	dN/dlogDp (number of particles per log of particle diameter)	PNSD	5 minutes	Once
Radon	18	P3 – Zagreb P4 - Filtration	Bq/m ³ (Bequerel per cubic meter)	Rn	72 hours - monthly	6 months
Microplastics	19	P3 – Zagreb P4 – Filtration C3 – Vilnius	#/cm ² (number per centimeter square)	MP	1 week	6 months

FFilter sampling have been and will be done across all pilots and campaigns listed above to determine the chemical components of aerosols collected through a filter. Because of a differing set-up, the three pilots will be reporting the parameters according to their respective time resolutions. For the following parameters, name and reference URI are referred to the European Chemical Agency portal (ECHA)¹⁴.

Table 5 Chemical components of indoor air

Parameter Name	ID	Common vs Pilot-specific	UoM	Symbol	Sampling resolution	Data sharing frequency
Volatile Organic Compounds - Aliphatic compounds						
Methylcyclopentane	20	P1 – Ferrara P3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₆ H ₁₂ (Methylcyclopentane)	15 days/6 hours	6 months
Cyclohexane	21	P1 – Ferrara P3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₆ H ₁₂ (Cyclohexane)	15 days/6 hours	6 months
2-Methylpentane	22	P1 – Ferrara P3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₆ H ₁₄	15 days/6 hours	6 months

¹⁴ <https://echa.europa.eu/home>



Heptane	23	P1 – Ferrara P3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₇ H ₁₄	15 days/6 hours	6 months
2-Methylhexane	24	P1 – Ferrara P3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₇ H ₁₆	15 days/6 hours	6 months
Volatile Organic Compounds - Chlorinated compounds						
Methylene chloride	25	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	CH ₂ Cl ₂	15 days/6 hours	6 months
Chloroform	26	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	CHCl ₃	15 days/6 hours	6 months
Trichloroethylene	27	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₂ HCl ₃	15 days/6 hours	6 months
Tetrachloroethylen e	28	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₂ Cl ₄	15 days/6 hours	6 months
1,4- dichlorobenzene	29	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₆ H ₄ Cl ₂	15 days/6 hours	6 months
Volatile Organic Compounds - Aromatic compounds						
Benzene	30	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₆ H ₆	15 days/6 hours	6 months
Toluene	31	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₇ H ₈	15 days/6 hours	6 months
Ethylbenzene	32	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₈ H ₁₀ (Ethylbe nzene)	15 days/6 hours	6 months



o-xylene	33	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₈ H ₁₀ (o-xylene)	15 days/6 hours	6 months
p-xylene	34	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₈ H ₁₀ (p-xylene)	15 days/6 hours	6 months
Styrene	35	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₈ H ₈	15 days/6 hours	6 months
1,3,5-trimethylbenzene	36	P1 – FerraraP3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	C ₉ H ₁₂	15 days/6 hours	6 months
Volatile Organic Compounds - Polycyclic aromatic hydrocarbons (PAHs) - determined from gravimetric PM1						
Fluoranthene	37	P3 – Zagreb P4 - Filtration	ng/m ³ (nanogram per cubic meter)	Flu	7 days	6 months
Pyrene	38	P3 – Zagreb P4 - Filtration	ng/m ³ (nanogram per cubic meter)	Pyr	7 days	6 months
Benzo[a]anthracene	39	P3 – Zagreb P4 - Filtration	ng/m ³ (nanogram per cubic meter)	BaA	7 days	6 months
Chrysene	40	P3 – Zagreb P4 - Filtration	ng/m ³ (nanogram per cubic meter)	Chry	7 days	6 months
Benzo[j]fluoranthene	41	P3 – Zagreb P4 - Filtration	ng/m ³ (nanogram per cubic meter)	BjF	7 days	6 months
Benzo[b]fluoranthene	42	P3 – Zagreb P4 - Filtration	ng/m ³ (nanogram per cubic meter)	BbF	7 days	6 months
Benzo[k]fluoranthene	43	P3 – Zagreb P4 - Filtration	ng/m ³ (nanogram per cubic meter)	BkF	7 days	6 months



Benzo[a]pyrene	44	P3 – Zagreb P4 - Filtration	ng/m ³ (nanogram per cubic meter)	BaP	7 days	6 months
Dibenzo[a,h]anthracene	45	P3 – Zagreb P4 - Filtration	ng/m ³ (nanogram per cubic meter)	DahA	7 days	6 months
Benzo(ghi)perylene	46	P3 – Zagreb P4 - Filtration	ng/m ³ (nanogram per cubic meter)	BghiP	7 days	6 months
Indeno[1,2,3-cd]pyrene	47	P3 – Zagreb P4 - Filtration	ng/m ³ (nanogram per cubic meter)	IP	7 days	6 months
Volatile Organic Compounds – Aldehyde compounds – New components added in D4.7						
Formaldehyde	48	P1 – Ferrara	µg/m ³ (microgram per cubic meter)	CH ₂ O	6 hours	6 months
Acetaldehyde	49	P1 – Ferrara	µg/m ³ (microgram per cubic meter)	C ₂ H ₄ O	6 hours	6 months
Acrylaldehyde	50	P1 – Ferrara	µg/m ³ (microgram per cubic meter)	C ₃ H ₄ O (Acrolein)	6 hours	6 months
Propionaldehyde	51	P1 – Ferrara	µg/m ³ (microgram per cubic meter)	C ₃ H ₆ O (Propanal)	6 hours	6 months
Butyraldehyde	52	P1 – Ferrara	µg/m ³ (microgram per cubic meter)	C ₄ H ₈ O (Butanal)	6 hours	6 months
Pentanal	53	P1 – Ferrara	µg/m ³ (microgram per cubic meter)	C ₅ H ₁₀ O	6 hours	6 months
Isovaleraldehyde	54	P1 – Ferrara	µg/m ³ (microgram per cubic meter)	C ₅ H ₁₀ O (Isopentanal)	6 hours	6 months



Glutaral	55	P1 – Ferrara	µg/m ³ (microgram per cubic meter)	C ₅ H ₈ O ₂	6 hours	6 months
Hexanal	56	P1 – Ferrara	µg/m ³ (microgram per cubic meter)	C ₆ H ₁₂ O	6 hours	6 months
Benzaldehyde	57	P1 – Ferrara	µg/m ³ (microgram per cubic meter)	C ₇ H ₆ O	6 hours	6 months
Volatile Organic Compounds – Other Compounds – New components added in D4.7						
Sulphur dioxide (air)	58	P3 – Zagreb P4 - Filtration	µg/m ³ (microgram per cubic meter)	SO ₂	6 hours	6 months
Ethanol	59	P3 – Zagreb P4 - Filtration	ppm (parts- per notation)	C ₂ H ₅ OH	6 hours	6 months
Methane	60	P3 – Zagreb P4 - Filtration	ppm (parts- per notation)	CH ₄	6 hours	6 months
Hydrogen	61	P3 – Zagreb P4 - Filtration	ppm (parts- per notation)	H ₂	6 hours	6 months
Ammonia	62	P3 – Zagreb P4 - Filtration	ppm (parts- per notation)	NH ₃	6 hours	6 months

3.3 Auxiliary Data

The following tables offer a set of auxiliary information related to the context of EDIAQI pilots or campaigns. Each table provides a list of properties to be collected and then harmonised towards the standard data model (e.g. SensorThings).

Table 6 lists possible sources of air pollution, both outdoor and indoor, which will aid in the analysis of the data. For outdoor sources, data may be derived from official / authoritative data portals (open data catalogues or geoportals) as spatial datasets.

Table 6 Auxiliary environmental conditions

Category	Subcategory
Outdoor sources	Traffic



	Industry Long-range transported pollution Domestic heating
Indoor sources	Outdoor air pollution Combustion activities Pets Clothes Dust resuspended by movement Cleaning agents Cigarette smoke/vape

Table 7 aims to provide simple information about the occupancy of indoor spaces where measurements are taken. Of course, occupancy may vary over time (hour-by-hour during a single day, or within a week) and in relation to the use of pilot building and the typologies of occupants. In EDIAQI, static information about occupancy is managed with average values about the occupants at single indoor space where digital or analogical sampling devices are installed.

Table 7 Auxiliary occupancy parameters

Category	subcategory
Value	Number of occupants per room where IAQ is monitored
Type	Kindergartners Students Teachers Patients Visitors Workers Household dwellers
Gender	Female Male All genders Irrelevant
Age of occupants	0-5 years old 5-10 10-18 18-25



	25-65 >65 years old
--	------------------------

Finally, Table 8 considers the physical properties of the places where measurements are taken. Some properties may refer to the entire pilot building while others are at the level of a single room or specific part of the building.

Table 8 Auxiliary building parameters

Category	subcategories/definitions	
Building general – general information about the building which may house one or more rooms with IAQ monitoring		
ID	Identifier of the building	
Name	Building name if it exists	
Address locator	Address of the building	
Location	Geographical coordinates of the building	
Use of the pilot building	Residential ¹⁵ - Collective residence - Individual residence - Residence for communities Industrial ¹⁶ Commerce and services ¹⁷ - Office - Public service - Trade	
Building specific – information about the specific room where IAQ is monitored		
Use of single room or building part	Residential	Living room Bedroom Kitchen Open space (kitchen and living room)
	Industrial	Office Laboratories

¹⁵ <https://inspire.ec.europa.eu/codelist/CurrentValue/residential>

¹⁶ <https://inspire.ec.europa.eu/codelist/CurrentValue/industrial>

¹⁷ <https://inspire.ec.europa.eu/codelist/CurrentValue/commerceAndServices>



	Commerce and services Office Laboratories School classroom Kindergarten Day care centre Sport facility Restaurant Subway station
Level	-1 (basement) 0 (ground level) 1 2 3 4 5 6
Floor area	Numeric in m ²
Glazed surface area	Numeric in m ²
Ventilation	None Manual Electric fan Air conditioning system Air purifier Central ventilation meeting
Sensor placement	Height from ground (meters) Distance from ceiling (meters) Horizontal distance to adjacent walls (meters) Description (on wall parallel to window, on wall where windows are)

Table 9 Auxiliary parameters (behaviours)

Parameter Name	ID	Common (C) vs PilotSpecific	UoM	Sampling resolution	Values and temp. unit
Opening/closure	54	Individual openings and	boolean - 1 = open	Irregular (individual)	Individual records



		closures (of windows or doors)	- 0 = closed		
Movement of people	55	Individual movements	boolean - 1 = people detected - 0 = no movement	Irregular (individual)	Individual records

3.3 Epidemiologic Data

This questionnaire is designed to explore the interplay between demographic factors, lifestyle choices, work environments, and health conditions, with a specific focus on indoor spaces and buildings. The goal is to understand how individual characteristics—such as age, smoking habits, and work hours—interact with health outcomes like asthma, allergies, and respiratory symptoms given the indoor air measurements as objective data. By collecting detailed information on demographics, lifestyle behaviours, and exposure to potential risk factors in indoor settings, this survey aims to identify patterns and correlations that could inform health interventions, workplace policies, and environmental improvements. This data-driven approach seeks to enhance our understanding of how indoor spaces affect well-being across diverse populations.

Table 10 Epidemiological questionnaire for large scale building studies

Category	Subcategories
Demographic information	
Age Group	Under 18 18-24 25-34 35-44 45-54 55-64 65 or older
Gender	Male Female



	Non-binary / Third gender Prefer not to say
Education Level (ISCED)	No formal education Primary education (ISCED 1) Lower secondary education (ISCED 2) Upper secondary education (ISCED 3) Post-secondary non-tertiary education (ISCED 4) Short-cycle tertiary education (ISCED 5) Bachelor's or equivalent level (ISCED 6) Master's or equivalent level (ISCED 7) Doctoral or equivalent level (ISCED 8)
Lifestyle and work environment	
Smoking Status	Never smoked Former smoker (quit more than a year ago) Former smoker (quit within the last year) Occasional smoker Regular smoker (daily or most days) Vapes or uses e-cigarettes Other (please specify)
Smoking Frequency (if current smoker)	Less than once per week A few times per week Daily (1-5 cigarettes) Daily (6-10 cigarettes) Daily (11-20 cigarettes) More than 20 cigarettes per day
Passive Smoke Exposure	Yes No
Type of Work	Office/Desk job Manual labor Service industry Health and safety



	Education Other (please specify)
Average Hours at Work per Week	Less than 20 hours 20-30 hours 30-40 hours More than 40 hours
Health conditions	
Doctor-diagnosed Conditions	Asthma Allergies (please specify) Chronic Obstructive Pulmonary Disease (COPD) None of the above
Symptoms in the Past 12 Months	Eye irritations Frequent colds Headaches None of the above

Table 11 Toxicological parameters

Parameter Name	ID	Common (C) vs Pilot specific	UoM	Sampling resolution
Primary DNA damage ¹⁸		P3 – Zagreb	% of tail DNA	Per child
Metals/element found in blood and plasma ¹⁹		P3 – Zagreb	µg/L or mg/L	Per child
Genomic instability ²⁰		P3 – Zagreb	evaluated by the micronucleus assay, expressed by several parameters	Per child

¹⁸ Primary DNA damage - detection of DNA strand breaks and alkali-labile sites (essentially any DNA modification that becomes a strand break under alkaline conditions)

¹⁹ Metal/elements found in blood and plasma - determination of the essential (e.g. Cu, Mg, I, etc.) and potentially toxic (e.g. As, Cd, Pb, etc.) elements levels

²⁰ Genomic instability - detection of chromosome fragments or whole chromosome(s) that have become separated from the main nuclei during mitosis, or fragments emerged during the DNA repair



3.4 Biological Data

Table 8.3 shows the biological parameters studied in the frame of the microbiome analysis in the mattress dust to determine the composition of bacterial and fungal communities.

Table 8.4 Biological parameters

Parameter Name	ID	Common (C) vs Pilot specific	UoM	Symbol	Sampling resolution ²¹	No of reads ²²
Bacterial taxa, 16S rRNA ²³		P3 – Zagreb	ASV (amplicon sequence variant) ²⁴		Per bed mattress per child	6 164 389
Fungal taxa, ITS		P3 – Zagreb	ASV (amplicon sequence variant)		Per bed mattress per child	3 232 715

²¹ Sampling resolution- samples- one sample per mattress per child (method vacuuming)

²² No of reads - number of sequence readings from Illumina/total number of sequence readings - the higher the number, the more detailed and better the data

²³ 16S rRNA gene - is sequenced for bacteria, ITS region - is sequenced for fungi (both because they are conserved and variable in species - so each species has a variant)

²⁴ ASV (amplicon sequence variant) - specific sequences identified in microbiota samples, microbiota taxa are accurately identified and compared through asv



Table 12 Lifestyle parameters collected

Category	Common (C) vs PilotSpecific	subcategory
Education	C	Elementary/middle school High school Bachelor's degree Master's degree Postgraduate/PhD study
Income	P3 – Zagreb	I don't have independent source of income < 1000 EUR 1000 – 1500 EUR 1500-2000 EUR 20000-25000 EUR 2500 – 3000 EUR >3000 EUR
Employment	C	Employed Unemployed Retired Student Household manager Maternity/parental leave Parent with a caregiver status
Living environment	P3 – Zagreb	Rural area Suburban area Urban area (city, build environment) Urban area (city, green environment)
Sources of information related to indoor air quality	C	No information Media and media campaigns (radio, TV, film, magazines) School/educational context Profession experience/workplace Personal interest Experiences of family members/friends/acquaintances Independent research (books, internet, social networks) Reading the products labels (cosmetics, cleaning products etc.) This project (EDIAQI) / Other



4. Technical interoperability

4.1 Standards

This chapter contains the technical references to software interfaces, protocols, operations, etc. required at server or client levels to view, access and share EDIAQI data.

The details of these technical references are based on previous EU projects (e.g. Stardust H2020²⁵, Mistral Telecom²⁶, GeoSmartCity CIP-PSP²⁷, CitiEnGov Interreg-CE²⁸) as well as on well-known specifications defined by ISO, CEN, OGC and other standardisation organisations and referenced by European Directives, such as INSPIRE (2007/2/CE)²⁹ and Open Data (2018/1024)³⁰.

In the following sub-sections, we present technical requirements for all types of services to share, view and access/download data. The technical requirements are presented as tables and are divided into three main sections, following the typical client-server architecture, with “client software sending a request for data to the server through the internet, and the server accepting the request, process it and deliver the data packets requested back to the client”³¹. Usually, server-side components provide “various functionalities, often called services [or application programming interfaces³²], such as sharing data or resources among multiple clients or performing computations for a client”³³.

The requirements listed below are divided by type of software component (client, server, interface):

²⁵ <https://stardustproject.eu/>

²⁶ <https://www.mistralportal.it/>

²⁷ <http://www.geosmartcity.eu/>

²⁸ <https://programme2014-20.interreg-central.eu/Content.Node/CitiEnGov.html>

²⁹ <https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX%3A32007L0002>

³⁰ <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32019L1024>

³¹ <https://www.w3schools.in/what-is-client-server-architecture>

³² <https://en.wikipedia.org/wiki/API>

³³ [https://en.wikipedia.org/wiki/Server_\(computing\)](https://en.wikipedia.org/wiki/Server_(computing))



- **client:** requirements related to client software (desktop or web) directly used by human beings to search/discover, view, access EDIAQI related data.
- **server:** requirements related to server components, to be made available at single pilot's level (as well as at the level of EDIAQI Data Platform).
- **interface:** requirements related to standard interfaces and protocols to be considered at client and/or server-side levels to guarantee that two software components (client and server) exchange information securely over the Internet with interoperable capabilities.

Standards and reference software solutions were fully described to EDIAQI partners during 4 webinars organised from June to October 2023. Presentations and YouTube videos are available on the [EDIAQI project web site](#). Annex I also contains a list of reference materials available:

- **Webinar 1 (June 21st, 2023)**
 - Introduction
 - Data Interoperability Examples
 - SensorThings
 - Endpoint and STA requests
 - FROST
 - STApplus extension
 - INSPIRE
- **Webinar 2 (July 21st, 2023)**
 - Introduction
 - FROST-Overview
 - FROST Deployment
 - FROST Docker Deployment
 - NiFi Ingestion
 - Semantics
- **Webinar 3 (September 29th, 2023)**
 - Introduction
 - JS web dashboards for STA



- Python QGIS plugin STA
- **Webinar 4 (October 27th, 2023)**
 - Introduction
 - CityGML
 - CityGML Software
 - 3DCityDB
 - 3DCityDB Tools QGIS

The following subsections contain information about possible standards to be considered for the sharing of EDIAQI dynamic data collected from sensors.

4.1.1 SensorThings API

From a technological perspective, this means that sensors are “accessible via a unique URL identifier and its functionalities (e.g. observation results) are addressable via standard HTTP operations (GET, POST, PUT, etc.), thus allowing access to sensor data with a RESTful API”³⁴ (RESTful API is an interface used by two computer systems to exchange information securely over the Internet).

In EDIAQI, the term “sensor” is conceived with a broad meaning also including hardware/software equipment and mobile devices used to automatically register data (measurements). These measurements are already part of the EDIAQI semantics described above (see Chapter 3) while others can be easily mapped against the data model classes. In EDIAQI, the reference standard to be considered for sharing dynamic data in a formalised and interoperable way is SensorThings APIs (STA)³⁵ standard, approved in 2016 by the Open Geospatial Consortium (OGC)³⁶.

The solution identified requires that compatible REST clients are setup at single pilot level, be able to communicate with the STA service and send the data in with HTTP calls (or HTTPS) in POST, PUT or GET operations, transmitting data in JSON payload.

³⁴ https://52north.github.io/sensor-web-tutorial/08_iot.html

³⁵ <https://www.ogc.org/standard/sensorthings/>

³⁶ <https://www.opengeospatial.org/>



STA is based on the OGC/ISO 19156:2011 and provides an open and unified framework to interconnect IoT sensing devices, data, and applications over the Web. It is an open standard addressing the syntactic interoperability and semantic interoperability of the Internet of Things. It complements the existing IoT networking protocols such as CoAP, MQTT, HTTP, 6LoWPAN. While the above-mentioned IoT networking protocols address the ability for different IoT systems to exchange information, STA addresses the ability of different IoT systems to use and understand the exchanged information.

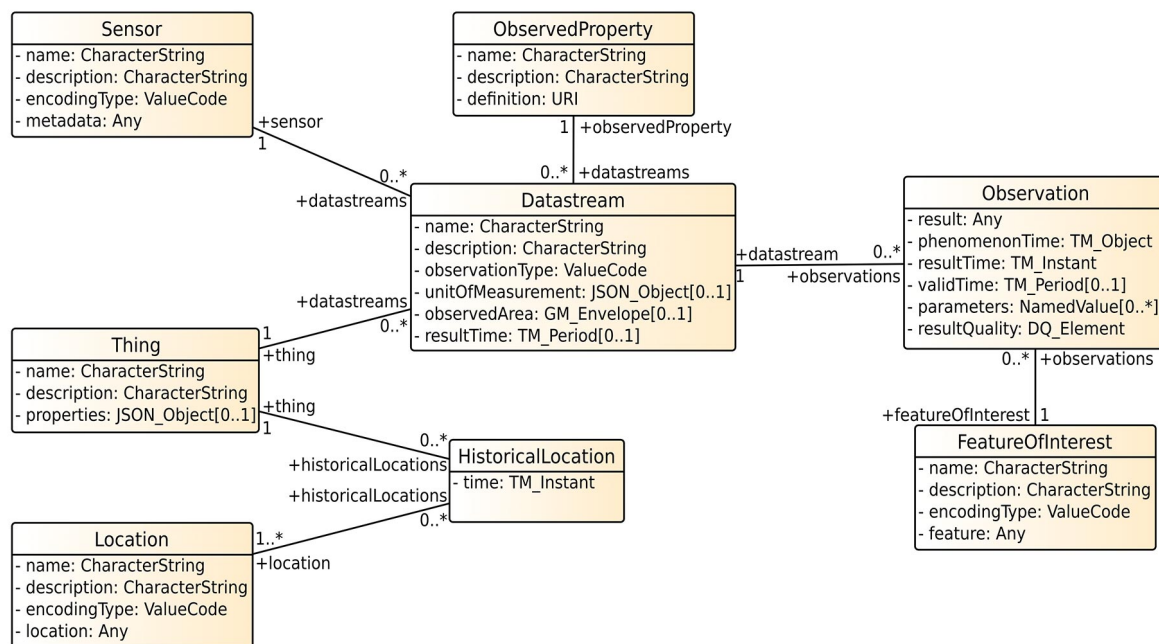


Figure 1 - Entities (classes) of the SensorThings data model

The meaning of STA entities must be very clear:

- **Thing**: an object of the physical world (physical thing) or the information world (virtual thing) that is capable of being identified and integrated into communication networks.
- **Locations**: locates the *Thing* or the *Things* it associated with.
- **HistoricalLocations**: provides the current (i.e., last known) and previous locations of the *Thing* with their time.
- **Datastream**: a collection of Observations and the Observations in a *Datastream* measure the same *ObservedProperty* and are produced by the same *Sensor*.
- **ObservedProperty**: specifies the phenomenon of an *Observation*.



- **Sensor:** an instrument that observes a property or phenomenon with the goal of producing an estimate of the value of the property.
- **Observation:** act of measuring or otherwise determining the value of a property.
- **FeatureOfInterest:** an *Observation* results in a value being assigned to a phenomenon. The phenomenon is a property of a feature, the latter being the *FeatureOfInterest* of the *Observation*.

As SensorThings is a RESTful web service, each entity can be CREATE, READ, UPDATE, and DELETE with standard HTTP verbs (POST, GET, PATCH, and DELETE).

It is also worth mentioning another standard defined by OGC: the SensorML³⁷ is a “*standard models and an XML encoding for describing sensors and measurement processes. SensorML can be used to describe a wide range of sensors, including both dynamic and stationary platforms and both in-situ and remote sensors*”.

The following tables provide the list of minimum requirements at client, server and interface levels:

Table 13 Technical details for sensor data streaming services - client

ID	DESCRIPTION
1.	Client-side software enabled to access dynamic data shall allow the users to specify a Bounding Box spatial filter in a map
2.	Client-side software shall allow the users to determine a time window filter with a (from, to) datetime fields.
3.	Client-side software shall allow the users to filter specific sensors (or in OGC terms, filtering by procedures or offerings).
4.	Client-side software shall allow the users to filter specific type of observation property.
5.	Client-side software shall allow the users to request existing observations for the content, spatial and temporal filters defined in previous requirements.
6.	Client-side software shall provide a mechanism to subscribe to streaming generated values defined with previously specified filters.
7.	Client-side software shall allow the user to explore the time series observation results for a single sensor.
8.	Client software shall show the available features of interest in a map

³⁷ <https://en.wikipedia.org/wiki/SensorML>

Table 14 Technical details for sensor data streaming services - server

ID	DESCRIPTION
9.	Server-side software shall support the OGC SensorThings API standard, with at least the following operations: • Create • Read • Update • Delete
10.	Server-side software shall support the following format bindings: • JSON • XML • KVP
11.	Server-side software shall provide a mean to subscribe to streaming data to avoid clients to develop a mechanism to decide whether new data is available.
12.	Server-side software shall be able to work with different RDBMS systems, with preference to open-source platforms as PostgreSQL.
13.	Server-side software shall provide a mechanism to connect the service to underlying MQTT standard protocol.

4.1.2 Web Map Service interface (ISO19128)

Web Mapping Services (WMS) provide the visualisation of geographical representation of real world's phenomena, i.e. a map, as picture of spatial objects. Geographical representation means producing and sharing an image (a map) from spatial data, applying a set of rendering rules, otherwise viewing a classic alphanumeric dataset can be achieved producing a tabular representation or a graphical one.

Due to the fact that sometimes geographical representation may be misunderstood as data access, it may be appropriate to highlight here the main differences:

1. accessing data involves the possibility of querying, sub-setting and filtering (it's a necessary condition, but not sufficient since certain view services have the capabilities of expressing a filter).
2. accessing data necessarily use a physical data format but does not depend on it; the representation of data instead is an integral part of viewing services.



3. very often in mapping services, the representation of data completely hides underlying data making it impossible to recover them (approximations, portrayal, simplifications, generalisation, aggregation, etc.).

In EDIAQI, maps and geographical representations of the locations of pilot buildings and monitoring units will be shown via standard protocols like the WMS (Web Map Service, also ISO19128) and/or the WMTS (Web Map Tile Service), both published by OGC. Tile based web maps will be made available based on storage of predefined tiles (from thousands to millions). The software components generated by EDIAQI will also offer functionalities to let clients visualise:

- tabular data, with filtering/searching capabilities to extract or sort subset of datasets.
- graphics (dashboards), based on open source Javascript libraries to render statistical data with high quality diagrams and presentation styles.

WMS provides a simple HTTP interface for requesting geo-registered map images from one or more distributed geospatial databases³⁸. A WMS request defines the geographic layer(s) and area of interest to be processed. The response to the request is one or more geo-registered map images (returned as JPEG, PNG, etc) that can be displayed in a browser application. The interface also supports the ability to specify whether the returned images should be transparent so that layers from multiple servers can be combined or not. Version 1.3 of WMS standard by OGC and ISO19128 are the same documents. The following tables provide the list of minimum requirements at client, server and interface levels:

Table 15 Technical details for web map services - client

ID	DESCRIPTION
1.	Client-side software enabled to view web maps shall allow users to view and browse a map (zoom, pan)
2.	Client-side software shall allow users to query and select data
3.	Client-side software shall allow users to retrieve maps and/or data provided by remote nodes, adding layers at least as OGC WMS
4.	Client-side software shall allow users to add layers to the map in the Table of Content (ToC) in the following ways:

³⁸ <https://www.ogc.org/standard/wms/>

	- from a URL (OWS GetCapabilities) - from a search on a list of catalogued services
5.	Client-side software shall allow users to change the opacity/transparency of each layer
6.	Client-side software shall allow users to export and save locally the context of the map in the following graphic formats: PNG, JPEG, PDF.
7.	Client-side software shall allow users to export and save locally the context of the map as hyperlink (permalink)
8.	Client-side software shall allow users to export and save locally vector data as KML file
9.	Client-side software shall allow users to export and save locally vector data as SHP file
10.	Client-side software shall allow users to change the Coordinate Reference System (CRS) of the map. The minimum set of CRSs available to the user shall be the one defined in INSPIRE Data Specifications.
11.	Client-side software shall allow users to change the CRS of the data exported (by means of Coordinate Transformation Service)
12.	Client-side software shall allow users to view metadata of selected WMS endpoints by accessing metadata through the WMS interface
13.	Client-side software shall allow users to view non-geographical information associated to layers, by clicking on the map (WMS GetFeatureInfo operation).

Table 16 Technical details for Map Services - server

ID	DESCRIPTION
14.	Server-side software shall support OGC WMS 1.3.0 (Web Map Service, also ISO19128)
15.	Server-side software shall support the following operations (OGC WMS): - GetCapabilities - GetMap - GetFeatureInfo
16.	Server-side software shall allow the loading and serving at least one of the following vector data formats: - GML (also according to INSPIRE Data Specifications) - ESRI Shapefile - JSON / Geojson - KML - CSV
17.	Server-side software shall support cascading Web Feature Service (WFS)



18.	Server-side software shall allow the representation of attributes values related to geographical phenomena as choropleth maps; choropleth maps are thematic maps in which areas are shaded or patterned in proportion to the measurement of the statistical variable being displayed on the map, such as population density or per-capita income
19.	Server-side software shall allow the representation of attributes values related to geographical phenomena as external graphics overlaid to the map

Table 17 Technical details for Map Services - interface

ID	DESCRIPTION
20.	Web mapping services shall comply with the “Commission Regulation (EC) No 976/2009 of 19 October 2009 implementing Directive 2007/2/EC of the European Parliament and of the Council as regards the Network Services”: http://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:32009R0976
21.	Web mapping services shall comply with INSPIRE View Service technical guidance: http://inspire.jrc.ec.europa.eu/documents/Network_Services/...
22.	Web mapping services shall support OGC SLD specification: http://www.opengeospatial.org/standards/sld
23.	Web mapping services shall support OGC Filter Encoding specification: http://www.opengeospatial.org/standards/filter
24.	Web mapping services shall support CQL specification: http://www.loc.gov/standards/sru/cql/spec.html

4.1.3 Web Feature Service interface (ISO19142)

As per INSPIRE definitions, a download service is equivalent to a service based on a standard protocol: in case of spatial data being managed as vectors, this means applying the Web Feature Service standard protocol (WFS, also ISO19142). The intention is that the user is given access to the raw data values instead of a cartographic representation (as is the case of mapping services with WMS requests that only return a map image as simple graphical representation).

In EDIAQI, WFS protocol is recommended to complement SensorThings API for static data (e.g. data about noise and sources of pollution, usually being spatial data and thus to make accessible through standard protocol).



WFS represents a change in the way geographic information is created, modified and exchanged on the Internet³⁹. Rather than sharing geographic information at the file level using File Transfer Protocol (FTP), for example, the WFS offers direct fine-grained access to geographic information at the feature and feature property level.

This International Standard specifies discovery operations, query operations, locking operations, transaction operations and operations to manage stored, parameterised query expressions.

Discovery operations allow the service to be queried to determine its capabilities and to retrieve the application schema that defines the feature types that the service offers.

Query operations allow features or values of feature properties to be retrieved from the underlying data store based upon constraints, defined by the client, on feature properties.

Locking operations allow exclusive access to features for the purpose of modifying or deleting features.

Transaction operations allow features to be created, changed, replaced and deleted from the underlying data store; stored query operations allow clients to create, drop, list and described parameterised query expressions that are stored by the server and can be repeatedly invoked using different parameter values.

In the taxonomy of services defined in ISO19119, WFS is primarily a feature access service but also includes elements of a feature type service, a coordinate conversion or transformation service and geographic format conversion service. Version 2.0.0 of WFS standard by OGC and the ISO19142 are the same documents.

Table 18 Technical details for Download Services - client

ID	DESCRIPTION
1.	Client-side software enabled to access geographical data shall allow users to extract features from download services and provide functionalities to download (subset of) datasets locally.
2.	Client-side software shall be able to compose queries (using CQL and/or OGC FE filters).
3.	Client-side software shall allow users to change the Coordinate Reference System (CRS) of the map. The minimum set of CRSs available to the user shall be the one defined in INSPIRE Data Specifications.

³⁹ <https://www.ogc.org/standard/wfs/>



Table 19 Technical details for Download Services – server

ID	DESCRIPTION
4.	Client-server software providing geographical data shall support OGC WFS 2.0 (Web Feature Service, ISO19142)
5.	Client-server software shall support operations (OGC WFS): • GetCapabilities • DescribeFeatureType • GetFeature
6.	Client-server software shall inform the client about the common and specific capabilities of a download access service.

Table 20 Technical details for Download Services - interface

ID	DESCRIPTION
7.	Client-server software shall comply with the “ Commission Regulation (EC) No 1088/2010 of 23rd of November 2010 amending Regulation (EC) No 976/2009 as regards download services and transformation services ”
8.	Client-server software shall comply with the INSPIRE Download Service technical guidance
9.	Client-server software shall support OGC FE specification
10.	Client-server software shall support CQL specification
11.	Client-server software shall support OGC GML specification (3.2.1)

4.1.4 CityGML

The CityGML standard defines a conceptual model and exchange format for the representation, storage and exchange of virtual 3D city models⁴⁰. It facilitates the integration of urban geodata for a variety of applications for Smart Cities, Urban Digital Twins and other use cases.

CityGML 3.0 is an evolution of the previous versions 1.0 and 2.0 of CityGML. While the previous versions standardised a GML exchange format, CityGML 3.0 standardises the underlying information model, and can be implemented in a variety of technologies

⁴⁰ <https://www.ogc.org/standard/citygml/>



including GML. The CityGML 3.0 Conceptual Model Standard describes a common semantic information model for the representation of 3D urban objects. The primary function of the model is to define the human interpretation of modelled data objects as well as their geometric representation and relationships. As such the CityGML 3.0 Conceptual Model provides a framework for integrating, storing, and exchanging 3D geospatial data.

CityGML 3.0 allows data to be encoded in GML/XML, but also in JSON or database schemas. Additional benefits compared to previous versions include much better integration with BIM, the ability to represent indoor spaces in different Levels of Detail (LOD), support for dynamic data provided by sensors and simulations, and the capability to extend the information model into Application Domain Extensions using Model Driven Architecture tools.

In the CityGML standard, the class “Building” is defined as *“a free-standing, self-supporting construction that is roofed, usually walled, and can be entered by humans and is normally designed to stand permanently in one place. It is intended for human occupancy (e.g. a place of work or recreation), habitation and/or shelter of humans, animals or things”*.

“A “Building” may be characterised by many “BuildingParts” (*a physical or functional subdivision of a Building. It would be considered a Building, if it were not part of a collection of other BuildingParts*) and/or many “BuildingRoom” (*a space within a Building or BuildingPart intended for human occupancy (e.g. a place of work or recreation) and/or containment of animals or things. A BuildingRoom is bounded physically and/or virtually*). Moreover, a building may have one or more “BuildingUnit” (*is a logical subdivision of a Building. BuildingUnits are formed according to some homogeneous property like function, ownership, management, or accessibility*).”

Different Level Of Details (LOD) may be considered for objects representing a building:



Figure 2 - Level Of Details (LOD) of CityGML



At LOD3 and LOD4, interior specific elements (spatial objects) can be modelled to represent single rooms, openings, windows and doors, with their specific properties.

4.2 Reference open-source software solutions

This section introduces a short list of reference software solutions implementing the standards mentioned above. All software listed below are open-source projects, free of charge and with source code released with open licenses. FROST server is the solution identified and suggested to efficiently implement an interoperability solution to share EDIAQI data (listed in chapter 0) among project partners. Open standards defined at international level guarantee the implementation of software solutions (in particular, open-source solutions) that implement them and facilitate the interoperability of data and web-based services.

4.2.1 FROST (FRaunhofer Opensource SensorThings-Server)

In EDIAQI, the FRaunhofer Opensource SensorThings-Server (FROST) solution is recommended to fully implement and exploit the capabilities of SensorThings standard. In chapter 0 a complete mapping is provided between the EDIAQI concepts (defined in chapter 0) and the standard data model of SensorThings. This will allow EDIAQI partners to share data about indoor air quality, outdoor air quality and auxiliary data in an interoperable a standard-conformant way.

The FRaunhofer Opensource SensorThings-Server is the first complete, open-source official reference implementation of the OGC SensorThings API Part 1: Sensing 1.0. In response to the increasing requirements arising from diverse applications within the "Internet of Things" paradigm, the Fraunhofer Institute for Optronics, System Technologies and Image Exploitation (IOSB) embarked on the development of a server tailored for the Standard SensorThingsAPI. The result of this effort is the FROST®-Server, an acronym denoting "Fraunhofer Open Source SensorThings API Server. As suggested by its name, FROST was developed to be open source to enhance usability within both research and commercial domains.

Moreover, the FROST Server is rooted in the SensorThings API and this foundation ensures adherence to widely recognised norms in spatial data representation.



FROST Server also provides versatile deployment alternatives, accommodating both Tomcat and Docker deployment frameworks. Notably, FROST demonstrates horizontal scalability, a feature integral to its adaptability. Additionally, leveraging its foundation on PostgreSQL, an inherently scalable relational database management system, FROST facilitates the establishment of multiple PostgreSQL clusters, augmenting its scalability in handling diverse workloads.

In the pursuit of optimal system performance, FROST Server is designed to meet the requirements of dynamic and data-intensive applications. This commitment to high performance aligns with the imperatives of contemporary data ecosystems, where rapid and efficient data processing is imperative for effective decision-making and operational continuity.

4.2.2 PostgreSQL / PostGIS

FROST Server relies on PostgreSQL⁴¹ solution to store and manage data about observations, observed properties, sensors, locations and other STA entities. Therefore, in EDIAQI PostgreSQL is recommended as RDBMS solution implementing SQL, with PostGIS extension enhancing the storage and management of spatial information (location of monitoring units, features of interest corresponding to pilot buildings and/or single rooms or buildings' parts) in compliance to ISO19100 standard series as well as OGC specifications.

PostGIS is an extension adding a spatial dimension to the capabilities of PostgreSQL. In the range of geospatial technology, PostGIS enriches PostgreSQL to store, query, and manipulate spatial data. It enables the integration of location-based information into database systems, offering a versatile solution for professionals in fields such as geography, urban planning, environmental science, and beyond.

4.2.3 Spatial server implementing WMS and WFS standard

In EDIAQI it is important to provide information about the “spatial” dimension of data. This means that at project level there is the need to deploy and configure a spatial server implementing WMS and WFS standards (mentioned above).

⁴¹ <https://www.postgresql.org/>



The solution recommended is GeoServer, “an open-source server written in Java that allows users to share, process and edit geospatial data. Designed for interoperability”⁴², GeoServer is able to manage a variety of data formats (including the mayor RDBMS platforms like Oracle, PostgreSQL/PostGIS or standard geographical formats like GML, Shapefile, GeoTIFF, ..) as well as remote sources. It provides access to data via standard OGC protocols (in particular, WMS and WFS) with different output formats (KML, GML, Shapefile, GeoRSS, PDF, GeoJSON, JPEG, GIF, SVG, PNG). GeoServer additionally supports efficient publishing of geospatial data to Google Earth using network links, using KML standard format (Google Earth)⁴³.

4.2.4 QGIS and plugin SensorThings API

To access and browse observations and other data provided within EDIAQI pilots and campaigns, a desktop GIS software could be a useful solution to fulfil the overall goal of making EDIAQI data really “FAIR” (Findable, Accessible, Interoperable and Reusable). The recommended GIS software is the open-source Quantum Geographic Information System (QGIS), world-wide used by millions of users⁴⁴ at different levels and for different purposes.

In 2022, DedaNext⁴⁵ developed a SensorThings API plugin for QGIS within the AIR-BREAK project⁴⁶. The plugin enables QGIS software to access dynamic data from sensors, using SensorThings API protocol. In 2024, the functionalities of this plugin will be refactored and extended within the core code of QGIS (embedding the capabilities to access SensorThings endpoints in the basic QGIS component, without the need to download and install an external plugin)⁴⁷.

⁴² <https://docs.geoserver.org/2.21.x/en/user/introduction/history.html>

⁴³ <https://ogc.org/standards/kml>

⁴⁴ <https://www.gispo.fi/en/blog/how-big-is-the-qgis-community/>

⁴⁵ <https://www.dedanext.it/>

⁴⁶ <https://www.uia-initiative.eu/en/uia-cities/ferrara>

⁴⁷ <https://github.com/qgis/QGIS-Enhancement-Proposals/issues/257>



4.2.5 3DCityDB + QGIS plugin 3DCityDB Tool

As EDIAQI data are related to “buildings” and indoor spaces, a 3D model of pilot buildings is suggested as possible goal to better and fully exploit the spatial dimension of EDIAQI data. In particular, a spatial 3D model of a building allows for the integration of heterogeneous (geo-related) information within a single framework and, therefore, create and manage complex spatial analysis.

To store 3D models, both file-based and database approaches can be used: there is no single, unique representation schema due to the heterogeneity and diversity of 3D models of spatial objects. The 3D City Database (3DCityDB) is suggested for modelling and storing spatially enabled information about EDIAQI pilot buildings; 3DCityDB a free and open-source package consisting of a database schema (based on PostgreSQL/PostGIS platform and Oracle Spatial) and a set of software tools to import, export, manage, analyse, and visualise virtual 3D models.

3DCityDB is built upon the CityGML standard and represents its SQL implementation. To fully exploit the 3DCityDB, another QGIS plugin is suggested: the 3DCityDB Tools for QGIS⁴⁸.

⁴⁸ <https://github.com/tudelft3d/3DCityDB-Tools-for-QGIS>



5. Mapping EDIAQI concepts towards standard data models

This section contains the mapping proposed between the data considered in EDIAQI (chapter 0) and standard data models (chapter 0).

5.1 STA data model

For the following data (listed in chapter 0), the mapping proposed is towards SensorThings.

The SensorThings data schema is based on the “[OGC SensorThings API Part 1: Detection](#)”:

- Indoor air quality parameters measured with sensorics.
- Environmental conditions recorded by sensorics.
- Outdoor air quality parameters measured by sensorics.
- Additional indoor air quality parameters measured in specific pilots and campaigns.
- Chemical components of indoor air.
- Auxiliary occupancy parameters.
- Auxiliary building parameters.
- Epidemiological data
- Biological data
- Lifestyle data

As an implementation of the Sensor Things API (STA), in chapter 0 the open-source FROST implementation is suggested, documented in the corresponding Frost Github repository⁴⁹.

The initial configuration of the FROST application also includes the creation of a schema compliant with the STA standard on the PostgreSQL database (describe above), therefore please refer to the FROST installation instructions available on Github.

In EDIAQI, the list of parameters to be loaded is mapped onto the entities of the SensorThings schema according to the following tables, where reference is made to the names of the attributes of the original structure of the data and to JSON Objects containing user-annotated properties as Key-Value Pairs, which is used to distinguish the data from different pilots and/or to provide further information and details (for instance, sensors’

⁴⁹ <https://github.com/FraunhoferIOSB/FROST-Server>



metadata⁵⁰). As anticipated in chapter 4, the SensorThings standard considers the following entities (sources: ISO19156 “Observation&Measurements”⁵¹ and W3C “Semantic Sensor Network Ontology”⁵²):

- **Thing:** a thing is any object of the physical world (physical things) or the IT world (virtual things) that is capable of being identified and integrated into communication networks [ITU-T Y.2060]. In EDIAQI, a Thing corresponds to the physical object equipped with one or more sensors to measure certain phenomena (ObservedProperties).
- **Location:** one example of location is the building or the room in which a wifi-connected thermostat is located⁵³. And the FeatureOfInterest of the Observations made by the thermostat (e.g., room temperature readings) should also be the building or the room. In EDIAQI, a Location corresponds to the geographical position (long/lat) where the Thing is installed (e.g. air quality monitoring unit).
- **FeatureOfInterest:** following the example above, the FeatureOfInterest of a wifi-connect thermostat can correspond to the Location of the thermostat (i.e., the living room where the thermostat is located). In EDIAQI, a feature of interest corresponds to the single room or part of the building whose properties are being measured (ObservedProperties).
- **Sensor:** a sensor is any entity capable of observing a phenomenon and returning an observed value as output. According to EEA EIONET vocabulary⁵⁴, different types of measurements can be considered for air quality (active, automatic, passive, remote, unknown). In EDIAQI, a sensor is the actual device (with electronic or mechanical

⁵⁰ SensorThings API defines two common Sensor metadata encodingTypes. Most sensor manufacturers provide their sensor datasheets in a PDF format. As a result, PDF is a Sensor encodingType supported by SensorThings API. The second Sensor encodingType is SensorML: <https://www.ogc.org/standard/sensorml/>

⁵¹ <https://docs.ogc.org/is/18-088/18-088.html>

⁵² <https://www.w3.org/TR/vocab-ssn/>

⁵³ STA data model also provides the concept.

⁵⁴ <http://dd.eionet.europa.eu/vocabulary/aq/measurementtype/view>



components) used to take the measurement, for instance related to air quality or other aspects.

- **ObservedProperty:** W3C SSN describes the “ObservableProperty” (synonym) as an observable quality (property, characteristic) of a FeatureOfInterest. Examples are the height of a tree, the depth of a water body, the temperature of a surface, or the concentration of NO₂ are examples of observable properties). In EDIAQI, an observed property refers to the single parameter measured.
- **Observation:** an Observation is the act of measuring or otherwise determining the value of a property⁵⁵. According to W3C “SSN”, it corresponds to the procedure to estimate or calculate a value of a property of a FeatureOfInterest; links to a Sensor to describe what made the Observation and how; links to an ObservableProperty to describe what the result is an estimate of, and to a FeatureOfInterest to detail what that property was associated with. In EDIAQI, an observation is the single resulting quantitative value and its correspondent datetime (expressed in ISO 8601 standard format)
- **DataStream:** this entity groups a collection of Observations measuring the same ObservedProperty and produced by the same Sensor. In EDIAQI, a datastream is the logical group of observations of a specific observed property together with a specific unit of measurement, made by a specific sensor.

In the following tables (one for each STA entity) a list of elements is provided as defined by the STA standard data model; specific data needed in EDIAQI will be implemented using the “properties” element and nested sub-elements. The tables contain, for each element, the corresponding datatype (e.g. string, integer, double, ...) and its multiplicity:

- 0..1 = optional (max one occurrence)
- 0..N = optional (more than one occurrence)
- 1 = mandatory (max one occurrence)

⁵⁵ OGC 10-004r3 and ISO 19156:2011



5.1.1 Things

In EDIAQI, a “Thing” corresponds to the physical object equipped with one or more sensors to measure certain phenomena (ObservedProperties). For simplicity, a “Thing” may be monitoring unit (equipped with more than one sensor) or a sampler.

Table 21 Mapping EDIAQI data towards STA data model: “Things” entity

STA element	Description (EDIAQI)	Data type	Multiplicity	Example
name	Name or code of the thing	String	1	EDIAQI.FE.001 for pilot P1 Ferrara
description	Verbose description of the thing	String	1	IAQ unit 001 at Istituto Copernico-Carpeggiani Ferrara for monitoring parameters A, B, C, D, ...
properties	A JSON Object containing user-annotated properties as key-value pairs	JSON Object	0..1	
identifier	Project id for the thing with the following pattern: - EDIAQI (fixed) - 2-digit code for pilot area (es. FE) - 3-digit code for the thing	String	1	EDIAQI.FE.001
owner	URI of the owner of the thing	URL	1	https://www.labservice.it/
beginTime	Datetime of installation of the thing	dateTime (YYYY-MM-DD)	1	2023-01-02



Examples of a “Thing” can be Radiello passive samplers (left) or NetPID monitoring units (right):



Figure 3 - Radiello passive sampler



Figure 4 - NetPID monitoring unit



This project has received funding from the European Union's Horizon Europe Framework Programme under grant agreement N° 101057497.

5.1.2 Locations

In EDIAQI, a Location corresponds to the geographical position where the Thing is installed (e.g. air quality monitoring unit). The location shall be provided at least as geographical coordinates of the pilot building, but it is preferred to provide the exact location of the single thing.

In both cases, the geometry representing the location of the thing is a “point”, with geographical coordinates expressed in longitude and latitude (spatial reference system WGS84 aka EPSG:4326). The “properties” element will allow to manage the specific information needed in EDIAQI to characterise the pilot building (as whole building) and the single room or building part where the thing is located, with details about its installation.

Table 22 Mapping EDIAQI data towards STA data model: “Locations” entity

STA element	Description (EDIAQI)	Data type	Multiplicity	Example
name	Name or code of the location expressed as: - name of the building (if available) - name or code of the single room or building part	String	1	Istituto Copernico-Carpeggiani Ferrara, classe 6A
description	Description of the location where the thing is located, with verbose details	String	1	The thing has been installed at the ground floor in classroom 6A of the high school Istituto Copernico-Carpeggiani
encodingType	Type of encoding for representing the location of the pilot building	Default = ‘application/geo+json’ ⁵⁶	1	application/geo+json

⁵⁶ Currently geojson is the only Location encodingType of the SensorThings API. GeoJSON supports the following geometry types: Point, LineString, Polygon, MultiPoint, MultiLineString, and MultiPolygon. Geometric objects with additional properties are Feature objects. Sets of features are contained by FeatureCollection objects:

<https://datatracker.ietf.org/doc/html/rfc7946>



location	(note: the following rows represent an excerpt of a normal GeoJSON structure for a point representing the exact location as longitude/latitude of the thing installed)		1	
type	Type of spatial feature representing the location of the thing	Default = 'Point'	1	Point
coordinates	List of coordinates	List of geographical coordinates	1	
0	First coordinate (long) of the location of the pilot building	Double (min. 5 decimals, max 8)	1	11.64273
1	Second coordinate (lat) of the location of the pilot building	Double (min. 5 decimals, max 8)	1	44.82996
properties	A JSON Object containing user-annotated properties as key-value pairs	JSON Object	0..1	
identifier	Project id for the location with the following pattern: - EDIAQI (fixed) - 2-digit code for pilot area (es. FE) - 3-digit code for the location of the thing	String	1	EDIAQI.FE.001
building	General information about the pilot building where the thing is located		1	
owner	URI of the owner of the pilot building	URL	0..1	https://www.provincia.fe.it/
identifier	Local identifier of the pilot building	String	0..1	FEIS01200X
address	Address locator, expressed as human readable designator or name that allows a user or application to reference and distinguish the address from neighbour addresses, within the scope of a thoroughfare name, address area name, administrative unit name or postal	String	1	Via Pontegradella 25, 44123 Ferrara



	designator, in which the address is situated			
country	Code of the country (ISO 3166-1) where the pilot building is located	String	1	IT
use	Current use of the whole building, according to INSPIRE Directive (2007/2/CE) codelist for “Buildings” data theme ⁵⁷	Codelist Residential - Collective residence - Individual residence - Residence for communities Industrial Commerce and services - Office - Public service - Trade	1	Public service
use_details	Detailed information about the use of the whole pilot building	String	0..1	Public high school
3Dmodel	External reference to 3D model representing the whole building as geographical dataset (e.g. in CityGML or KML format)	URL	0..1	
room	Details about the placement of the thing (sensor placement in §Auxiliary Data)		0..1	
identifier	Local identifier of the room	String	0..1	6A
elevation	Overall height above sea level (meters) of the thing	Double (2 decimals)	0..1	14.23
level	Floor number where the thing is located (single room or building part)	Integer	0..1	3

⁵⁷ <https://inspire.ec.europa.eu/codelist/CurrentUseValue>



height	Height from room floor (meters) of the thing	Double (2 decimals)	0..1	1.55
d_ceiling	Distance from room ceiling (meters) of the thing	Double (2 decimals)	0..1	3.12
d_horizontal	Horizontal distance to adjacent walls (meters) of the thing	Double (2 decimals)	0..1	0.90
3Dmodel	External reference to 3D model representing the single room (e.g. in CityGML or KML format)	URL	0..1	





Figure 5 - Location of the Thing corresponding to building address ([see it on GoogleMaps](#))



Figure 6 - Exact location of the Thing ([see it on GoogleMaps](#))

5.1.3 FeatureOfInterest

In EDIAQI, a feature of interest strictly corresponds to the single room or part of the building whose properties are being measured (ObservedProperties). The element “properties” can be used to manage specific details needed in EDIAQI to manage information related to the room or building part and its occupancy.



This project has received funding from the European Union’s Horizon Europe Framework Programme under grant agreement N° 101057497.

Table 23 Mapping EDIAQI data towards STA data model: “FeatureOfInterest” entity

STA element	Description (EDIAQI)	Data type	Multiplicity	Example
Name	Name or code of the single room or building part that is monitored	String	1	Class6A
description	Verbose description of the feature of interest (single room or building part)	String	1	Classroom 6A
encodingType	Type of encoding for representing the feature of interest (single room or building part)	Default = ‘application/geo+json’ ⁵⁸	1	application/geo+json
Feature	(note: the following rows represent an excerpt of a normal GeoJSON structure for a polygon with N vertexes)		1	
type	Geographical extent of the area (single room or building part) whose properties are being measured	Default = ‘Polygon’	1	Polygon
coordinates	List of coordinates	List of geographical coordinates	1	
0	1st vertex of the polygon			
0	Longitude of the feature of interest (single room or building part)	Double (min. 5 decimals, max 8)	1	11.64273
1	Latitude of the feature of interest (single room or building part)	Double (min. 5 decimals, max 8)	1	44.82996
1	2nd vertex of the polygon			

⁵⁸ Currently geojson is the only Location encodingType of the SensorThings API. GeoJSON supports the following geometry types: Point, LineString, Polygon, MultiPoint, MultiLineString, and MultiPolygon. Geometric objects with additional properties are Feature objects. Sets of features are contained by FeatureCollection objects (<https://datatracker.ietf.org/doc/html/rfc7946>)



0	Longitude of the feature of interest (single room or building part)	Double (min. 5 decimals, max 8)	1	11.64273
1	Latitude of the feature of interest (single room or building part)	Double (min. 5 decimals, max 8)	1	44.82996
N	Nth vertex of the polygon			
0	Longitude of the feature of interest (single room or building part)	Double (min. 5 decimals, max 8)	1	11.64273
1	Latitude of the feature of interest (single room or building part)	Double (min. 5 decimals, max 8)	1	44.82996
properties	A JSON Object containing user-annotated properties as key-value pairs	JSON Object	0..1	
identifier	Project id for the feature of interest, with the following pattern: - EDIAQI (fixed) - 2-digit code for pilot area (es. FE) - 3-digit code for the feature of interest	String	1	EDIAQI.FE.001
Room	Details about the placement of the thing (sensor placement in \$Auxiliary Data)		0..1	
identifier	Local identifier of the single room or building part	String	0..1	6A
elevation	Overall height above sea level (meters) of the thing	Double (2 decimals)	0..1	14.23
level	Floor number where the thing is located (single room or building part)	Integer	0..1	0
3Dmodel	External reference to 3D model representing the single room (e.g. in CityGML or KML format)	URL	0..1	
Use	Current use of the feature of interest (single room or building part)	Codelist - Living room - Bedroom - Kitchen	1	Classroom



This project has received funding from the European Union's Horizon Europe Framework Programme under grant agreement N° 101057497.

		- Open space (kitchen and living room) - Office - Laboratory - Classroom - Sport facility - Restaurant - Subway station		
use_details	Detailed information about the use of the feature of interest (single room or building part)	String	0..1	Classroom used by teachers and students for daily lessons
floor_area	Area (in m ²) of the floor of the feature of interest (single room or building part)	Integer	0..1	120
glazed_area	Area (in m ²) of the glazed surface (windows) of the feature of interest (single room or building part)	Integer	0..1	15
ventilation	Presence of ventilation system in the single room or building part	Codelist: - None - Manual - Electric fan - Air conditioning system - Air purifier - Central ventilation meeting	0..1	Air conditioning system
occupants ⁵⁹	Information about the occupants of the single room or building part		0..N	
number	Number of occupants	Integer	1	22
type	Typology of occupants	Codelist: - Kindergartners - Students - Teachers	1	Students

⁵⁹ The element “occupants” can be repeated more than once, with different typologies.



		- Patients - Visitors - Workers - Household dwellers		
gender	Gender of occupants (per typology)	Codelist: - Female - Male - All genders - Irrelevant	0..1	All genders
age	Age of occupants (per typology)	Codelist: - under 5 years old - 5-10 - 10-18 - 18-25 - 25-65 - over 65 years old	0..1	10-18



The polygon representing the feature of interest will be composed by an array with vertices having geographical coordinates (longitude, latitude in WGS84) as in the following example (green polygon):



Figure 7 - Example of FeatureOfInterest corresponding to a single room

In the GeoJSON Object representing the feature, the spatial information about the polygon will be managed as in the following excerpt (corresponding to the green polygon):

```
{
  "type": "FeatureCollection",
  "name": "temp",
  "crs": { "type": "name", "properties": { "name":
    "urn:ogc:def:crs:OGC:1.3:CRS84" } },
  "features": [
    { "type": "Feature", "properties": { },
      "geometry": { "type": "Polygon", "coordinates": [
        [ [ 11.643365834152107, 44.830455040411074 ], [
          11.643441802829082, 44.830347878077305 ], [
          11.643290387040651, 44.830294443474003 ], [
          11.643215296187936, 44.830402191139434 ], [
          11.643365834152107, 44.830455040411074 ] ] ] } }
  ]
}
```



5.1.4 Sensors

In EDIAQI, a “Sensor” is the actual device (with electronic or mechanical components) used to take the measurement, for instance related to air quality or other aspects. Each “Sensor” will be detailed in the “metadata” element (as URL to online document containing technical specifications) and it will be characterised by specific details contained in the element “properties”, with external reference to official EIONET vocabularies.

Table 24 Mapping EDIAQI data towards STA data model: “Sensors” entity

STA element	Description (EDIAQI)	Data type	Multiplicity	Example
name	Name or code of the sensor that is used to monitor a specific parameter	String	1	Sensirion SPS30
description	Description of the sensor	String	1	PM2.5 Sensor for HVAC and air quality applications SPS30
encodingType	Type of encoding for representing technical details of the sensor	Default = ‘application/pdf’ ⁶⁰	1	application/pdf
metadata	Technical details of about the sensor	URI	1	https://sensirion.com/media/documents/B7AAA101/61653FB8/Sensirion Particulate Matter AppNotes Specification Statement.pdf
properties	A JSON Object containing user-annotated properties as key-value pairs	JSON Object	0..1	

⁶⁰ SensorThings API defines two common Sensor metadata encodingTypes. Most sensor manufacturers provide their sensor datasheets in a PDF format. As a result, PDF is a Sensor encodingType supported by SensorThings API. The second Sensor encodingType is SensorML: <https://www.ogc.org/standard/sensorml/>.



identifier	Project id for the sensor with the following pattern: - EDIAQI (fixed) - 2-digit code for pilot area (es. FE) - 3-digit code for the sensor	String	1	EDIAQI.FE.001
measurementType	URI of the type of measurement	URL of the EIONET Codelist "measurementType" ⁶¹ with the exact type of measurement	1	http://dd.eionet.europa.eu/vocabulary/aq/measurementtype/automatic
method	URI of the type of method	EIONET Codelist "measurementMethod" ⁶² with the exact type of method	1	http://dd.eionet.europa.eu/vocabulary/aq/measurementmethod/NDIR
owner	URI of the owner of the sensor	URL	0..1	https://www.labservice.it/
responsibleParty	Contact details of the organization responsible for the sensor	JSON Object	0..1	
organizationName	Name of the organization	String	1	Lab Service Analytica srl
website	Website of the organization	String	0..1	https://www.labservice.it/
telephoneVoice	Telephone voice number of the organization (functional)	String	0..1	+39051732351
electronicMailAddress	Email of the organization (functional)	String	1	info@labservice.it

⁶¹ <http://dd.eionet.europa.eu/vocabulary/aq/measurementtype/>

⁶² <http://dd.eionet.europa.eu/vocabulary/aq/measurementmethod/>



Examples of a “Sensor” can be a Sensirion SPS30 (left) or NetPID monitoring units (right):

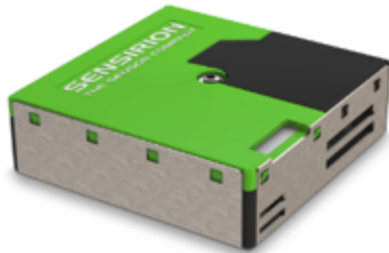


Figure 8 - Sensirion SPS30 for PM2.5



Figure 9 - MH-410D NDIR Infrared CO2



This project has received funding from the European Union's Horizon Europe Framework Programme under grant agreement N° 101057497.

5.1.5 ObservedProperties

In EDIAQI, an observed property refers to the single parameter measured, with external reference to official EIONET vocabulary.

Table 25 Mapping EDIAQI data towards STA data model: “ObservedProperties” entity

STA element	Description (EDIAQI)	Data type	Multiplicity	Example
name	Name or code of the sensor that is used to monitor a specific parameter	String	1	O3
definition	Definition of the observed property. For AQ parameters, refer to EIONET Codelist	EIONET Codelist “Pollutant” ⁶³	1	http://dd.eionet.europa.eu/vocabulary/aq/pollutant/7
description	Description of the sensor	String	1	Ozone

⁶³ <http://dd.eionet.europa.eu/vocabulary/aq/pollutant/>



5.1.6 Observations

In EDIAQI, an observation is the single resulting quantitative value and its correspondent datetime (expressed in ISO 8601 standard format)⁶⁴.

Table 26 Mapping EDIAQI data towards STA data model: “Observations” entity

STA element	Description (EDIAQI)	Data type	Multiplicity	Example
phenomenonTime	In EDIAQI, it corresponds to the period of time or to the exact instant related to the observation of a specific parameter (depending on the nature of the observed property)	TM_Object (ISO 8601) as Time Interval (e.g. for average hourly values) OR TM_Object (ISO 8601) as Time Instant (e.g. for single measurements)	1	2023-11-30T23:00:00Z/2023-12-01T00:00:00Z OR 2023-11-30T23:00:05Z
result	The value of an EDIAQI parameter (ObservedProperty) related to a specific period of time	Double (max 5 decimals)	1	12.34567
resultTime	The time of the Observation’s result was generated (i.e. inserted in the database and made available).	TM_Object (ISO 8601) as Time Instant	1	2023-12-15T00:00:00Z

⁶⁴ <https://www.iso.org/iso-8601-date-and-time-format.html>



5.1.7 DataStreams

In EDIAQI, a datastream is the logical group of observations of a specific observed property together with a specific unit of measurement (referencing the official EIONET vocabulary), made by a specific sensor.

Table 27 Mapping EDIAQI data towards STA data model: “DataStreams” entity

STA element	Description (EDIAQI)	Data type	Multiplicity	Example
name	Name or code of the collection of observations related to the same parameter in a single room or building part.	String	1	Copernico_6A_Ozone
description	Description of the collection of observations	String	1	Ozone at classroom 6A, Istituto Copernico-Carpeggiani in Ferrara
unitOfMeasurement	A JSON Object containing three key-value pairs. The name property presents the full name of the unitOfMeasurement; the symbol property shows the textual form of the unit symbol; and the definition contains the URI defining the unitOfMeasurement.	JSON Object	1	
name	Name of the unit of measure	Codelist from EIONET Vocabulary “Concentration” ⁶⁵	1	Micrograms per cubic meter

⁶⁵ <https://dd.eionet.europa.eu/vocabulary/eurostat/unit/>



symbol	Symbol of the unit of measure	Codelist from EIONET Vocabulary "Concentration" ⁶⁶	1	ug.m-3
definition	URI of the unit of measure	URL of the Codelist from EIONET Vocabulary "Concentration" ⁶⁷ with the exact type of concentration	1	http://dd.eionet.europa.eu/vocabulary/uom/concentration/ug.m-3
observationType	The type of Observation (with unique result type), which is used by the service to encode observations.	Default = "OM_Measurement"	1	http://www.opengis.net/def/observationType/OGC-OM/2.0/OM_Measurement
properties	A JSON Object containing user-annotated properties as key-value pairs	JSON Object	0..1	
identifier	Project id for the datastream with the following pattern: - EDIAQI (fixed) - 2-digit code for pilot area (es. FE) - 3-digit code for the datastream	String	1	EDIAQI.FE.001

⁶⁶ <https://dd.eionet.europa.eu/vocabulary/eurostat/unit/>

⁶⁷ <http://dd.eionet.europa.eu/vocabulary/uom/concentration/>



5.2 Examples of ‘insert’ operations

According to the standard SensorThings API, to create an entity in a the “Things” collection, a client SHALL send a HTTP POST request to that collection's URL. The POST body SHALL contain a single valid entity representation. If the target URL for the collection is a navigationLink, the new entity is automatically linked to the entity containing the navigationLink. Upon successful completion, the response SHALL contain a HTTP location header that contains the selfLink of the created entity.

To insert entities in SensorThings API, the commands shall follow the [guidelines available related to integrity constraints](#).

Table 28 Sensor Things API - Guidelines available related to integrity constraints

Scenario	Integrity Constraints
Create a Thing entity	-
Create a Location entity	[*]
Create a Sensor entity	-
Create an ObservedProperty entity	-
Create a FeatureOfInterest entity	-
Create a Datastream entity	SHALL link to a Thing entity.
	SHALL link to a Sensor entity
	SHALL link to an ObservedProperty entity.
Create an Observation entity	SHALL link to a Datastream entity.
	SHALL link to a FeatureOfInterest entity. If no link specified, the service SHALL create a FeatureOfInterest entity from the content of the Location entities.



[*] Location has not an integrity constraint forcing it to link to a specific Thing as the relation between Location and Things is of the type “many to many”.

However, the data model described in this document requires a relation linking a Location to the corresponding Thing and this will be present in the following examples.

For the sake of clarity, the following examples describe how to insert each type of STA Entity (Sensors, Things, Locations, etc) separately, but a more efficient approach could be to insert them together in a single call, as described in the [guidelines available](#).

5.2.1 Step 1. Create a Thing entity

POST <https://<domain>/FROST-Server/v1.1/Things>

```
{
  "@iot.id": "THI.FE.021",
  "name": "NetPID unit num. LS0623020143",
  "description": "NetPID EDIAQI unit installed in Istituto N.Copernico A.Carpeggiani (classroom 10) (Ferrara) for monitoring indoor AQ parameters (PM10, PM2.5, NO2, O3, CO, CO2, VOC) and indoor climate conditions (T, RH, Pressure).",
  "properties": {
    "identifier": "THI.FE.021",
    "owner": "https://www.labservice.it/",
    "beginDatetime": "2023-11-28 00:00:00",
    "endDatetime": null,
    "localIdentifier": "LS0623020143"
  }
}
```

To view the Thing: [https://<domain>/FROST-Server/v1.1/Things\('THI.FE.021'\)](https://<domain>/FROST-Server/v1.1/Things('THI.FE.021'))

5.2.2 Step 2. Create a Location entity

POST <https://<domain>/FROST-Server/v1.1/Locations>

```
{
  "@iot.id": "LOC.FE.021",
  "name": "Istituto N.Copernico A.Carpeggiani (classroom 10)",
  "description": "Classroom 10 at ground floor",
  "properties": {
    "identifier": "LOC.FE.021",
    "isOutdoor": false,
    "owner": "http://www.iiscopernico.edu.it/",
    "idOfTheBuildingInOpenStreetMap": null,
    "address": "Via Pontegradella, 25, 44123 Ferrara FE",
    "country": "IT",
    "useOfTheBuilding": "Public service",
    "detailsAboutTheUse": "High school, graphic design laboratory",
    "localIdentifier": "LS0623020143",
    "room": {
      "idOfTheRoom": null,
      "elevationOfTheThing": 5.766697,
      "levelFloor": 0,
      "verticalDistanceFromPavementOrGround": 0,
      "verticalDistanceFromCeilingOfTheRoom": 0,
      "horizontalDistanceFromWalls": 0
    }
  },
  "encodingType": "application/geo+json",
}
```



```
"location": {
  "type": "Point",
  "coordinates": [
    11.6429187,
    44.82984046
  ]
}
```

To view the Location: [https://<domain>/FROST-Server/v1.1/Locations\('LOC.FE.021'\)](https://<domain>/FROST-Server/v1.1/Locations('LOC.FE.021'))

5.2.3 Step 3. Create a Sensor entity

POST <https://<domain>/FROST-Server/v1.1/Sensors>

```
{
  "@iot.id": "SEN.001",
  "name": "NetPID EDIAQI indoor",
  "description": "Multi-sensor monitoring unit for indoor air quality assessment",
  "encodingType": "application/pdf",
  "metadata": "https://www.labservice.it/wp-content/uploads/2024/03/Data-Sheet-NetPID-EDIAQI-REV2.pdf",
  "properties": {
    "identifier": "SEN.001",
    "measurementType":
"http://dd.eionet.europa.eu/vocabulary/aq/measurementtype/remote",
    "method": "http://dd.eionet.europa.eu/vocabulary/aq/measurementtype/remote",
    "owner": "https://www.labservice.it/",
    "responsibleParty": {
      "organisationName": "Lab Service Analytica srl",
      "website": "https://www.labservice.it/",
      "telephoneVoice": "+39051732351",
      "electronicMailAddress": "info@labservice.it"
    }
  }
}
```

To view the Sensors: [https://<domain>/FROST-Server/v1.1/Sensors\('SEN.001'\)](https://<domain>/FROST-Server/v1.1/Sensors('SEN.001'))

5.2.4 Step 4. Create an Observed Property entity

POST <https://<domain>/FROST-Server/v1.1/ObservedProperties>

```
{
  "@iot.id": 6,
  "name": "O3",
  "description": "Ozone",
  "definition": " http://dd.eionet.europa.eu/vocabulary/aq/pollutant/7"
}
```



To view the ObservedProperty: [https://<domain>/FROST-Server/v1.1/ObservedProperties\(6\)](https://<domain>/FROST-Server/v1.1/ObservedProperties(6))

5.2.5 Step 5. Create a FeatureOfInterest entity

POST <https://<domain>/FROST-Server/v1.1/FeaturesOfInterest>

```
{
  "name": "Istituto N.Copernico A.Carpeggiani (classroom 10)",
  "description": "Classroom 10 at ground floor",
  "encodingType": "application/geo+json",
  "feature": {
    "type": "Point",
    "coordinates": [
      11.6429187,
      44.82984046
    ]
  },
  "properties": {
    "identifier": "FOI.FE.021",
    "localIdentifier": "LS0623020143",
    "room": {
      "indoorHeightOfTheRoom": null,
      "levelFloor": 1,
      "useOfTheSingleRoom": "Classroom",
      "useDetails": "Classroom",
      "floorArea": null,
      "glazedArea": null,
      "ventilation": "Manual"
    },
    "occupants": {
      "numberOfOccupants": null,
      "mainTypeOfOccupants": "Students",
      "genderOfOccupants": "All genders",
      "averageAgeOfOccupants": "10 to 18",
      "occupancyHoursMon": null,
      "occupancyHoursTue": null,
      "occupancyHoursWed": null,
      "occupancyHoursThu": null,
      "occupancyHoursFri": null,
      "occupancyHoursSat": null,
      "occupancyHoursSun": null
    }
  }
}
```

To view the FeaturesOfInterest: [https://<domain>/FROST-Server/v1.1/FeaturesOfInterest\('9935'\)](https://<domain>/FROST-Server/v1.1/FeaturesOfInterest('9935'))



5.2.6 Step 6. Create a Datastream entity

POST <https://<domain>/FROST-Server/v1.1/Datastreams>

```
{
  "id": "DST.FE.205",
  "method": "post",
  "url": "Datastreams",
  "body": {
    "Thing": {
      "@iot.id": "THI.FE.021"
    },
    "Sensor": {
      "@iot.id": "SEN.001"
    },
    "ObservedProperty": {
      "@iot.id": 6
    },
    "name": "O3@THI.FE.021",
    "description": "Ozone (air)",
    "unitOfMeasurement": {
      "name": "microgram per cubic meter",
      "symbol": "µg/m³",
      "definition": "https://dd.eionet.europa.eu/vocabularyconcept/aq/pollutant/7"
    },
    "observationType": "OM_Measurement",
    "properties": {
      "identifier": "DST.FE.205"
    }
  }
}
```

To view the Datastream: [https://<domain>/FROST-Server/v1.1/Datastreams\(205\)](https://<domain>/FROST-Server/v1.1/Datastreams(205))



5.2.7 Step 7. Create an Observation entity

POST <https://<domain>/FROST-Server/v1.1/Observations>

```
{
  "Datastream": {
    "@iot.id": 1
  },
  "phenomenonTime": "2023-07-04T22:00:00Z/2023-07-04T22:05:00Z",
  "result": 0.14934,
  "resultTime": "2023-07-04T22:20:00Z"
}
```

To view the Observation: [https://<domain>/FROST-Server/v1.1/Observations\(389920465\)](https://<domain>/FROST-Server/v1.1/Observations(389920465))

What follows is a more efficient approach to inserting bulk group of observations:

[https://<domain>/FROST-Server/v1.1/CreateObservations?\\$resultFormat=dataArray](https://<domain>/FROST-Server/v1.1/CreateObservations?$resultFormat=dataArray)

```
[
  {
    "Datastream": {
      "@iot.id": 7
    },
    "components": [
      "phenomenonTime",
      "result",
      "resultTime"
    ],
    "dataArray": [
      [
        "2023-07-04T22:00:00Z/2023-07-04T22:05:00Z",
        0.15132,
        "2024-04-30T13:00:00Z"
      ],
      [
        "2023-07-04T22:05:00Z/2023-07-04T22:10:00Z",
        0.14934,
        "2024-04-30T13:00:00Z"
      ],
      [
        "2023-07-04T22:10:00Z/2023-07-04T22:15:00Z",
        0.14938,
        "2024-04-30T13:00:00Z"
      ],
      [
        "2023-07-04T22:15:00Z/2023-07-04T22:20:00Z",
        0.15,
        "2024-04-30T13:00:00Z"
      ]
    ]
  }
]
```



5.3 Other standard data models to be considered

5.3.1 CityGML “Building” class

As aforementioned, different Level Of Details (LOD) may be considered for objects representing a building (see Figure 2 in chapter 0). At LOD3 and LOD4, interior specific elements (spatial objects) can be modelled to represent single rooms, openings, windows, doors, ... with their specific properties.

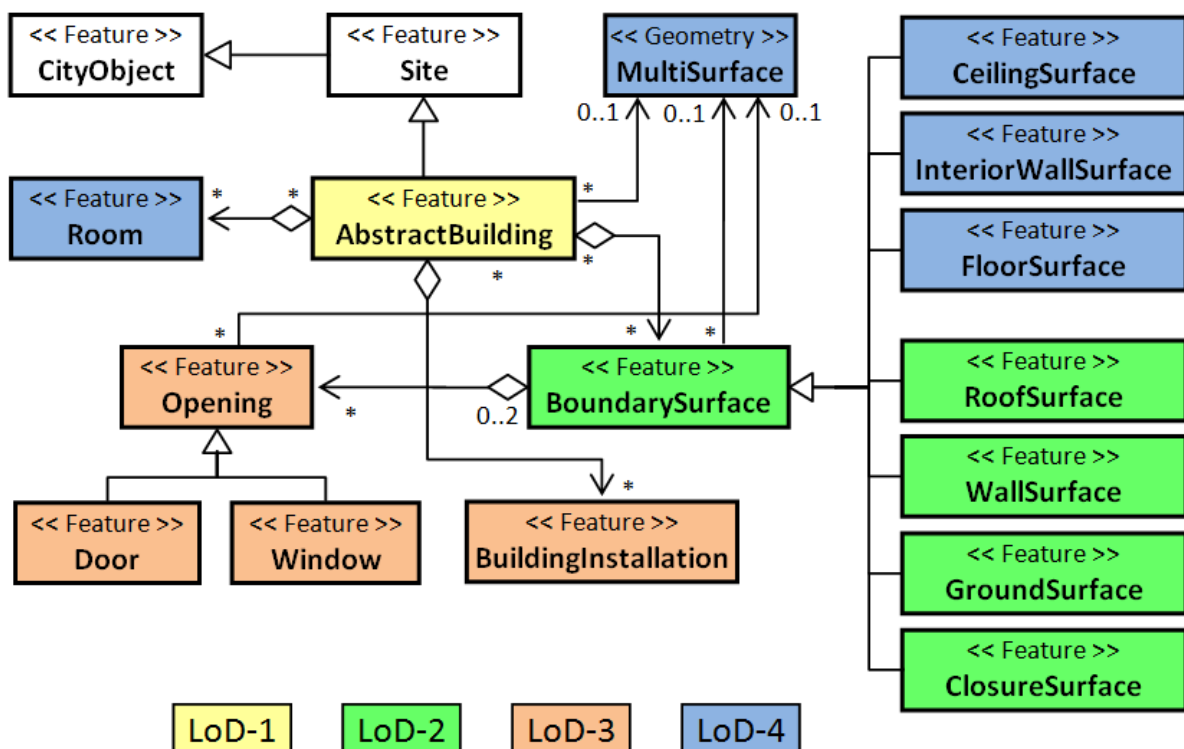


Figure 10 - Simplified UML Class Diagram for “Building” in CityGML

In future versions of this deliverable (i.e. deliverable D4.7 planned at M24) we will map existing data about pilot buildings in EDIAQI towards the CityGML standard data model.

5.3.2 INSPIRE Directive “Building” data theme

Among the 34 data themes considered by the European Directive 2007/2/CE (aka INSPIRE⁶⁸) the theme “Buildings” may be considered another target data model. The base UML Class Diagram for data representing buildings is the following:

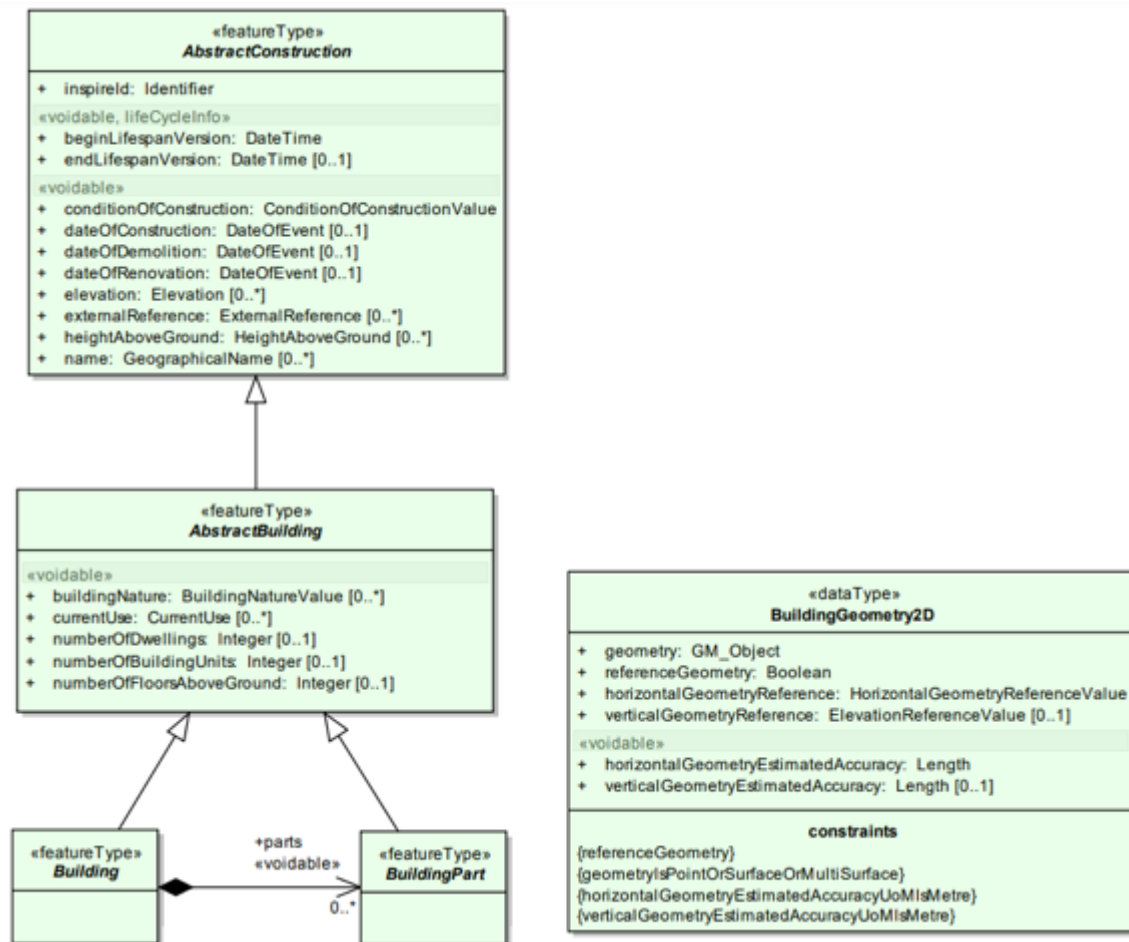


Figure 11 - Overview of the Building Base in INSPIRE Technical Guidelines (TG)⁶⁹

Indeed, the rooms and building parts may be represented by a solid or a multisurface, according to the various LoDs of CityGML.

Moreover, the components of the building may also be semantically described by specific feature types:

- in LoD2: boundary surfaces (e.g. walls and roof) and installations...

⁶⁸ https://knowledge-base.inspire.ec.europa.eu/index_en

⁶⁹ <https://github.com/INSPIRE-MIF/technical-guidelines/tree/main/data/bu>



- in LoD3: openings (doors and windows) are added.
- in LoD4: the interior of building (building units, rooms, internal installations) is added.

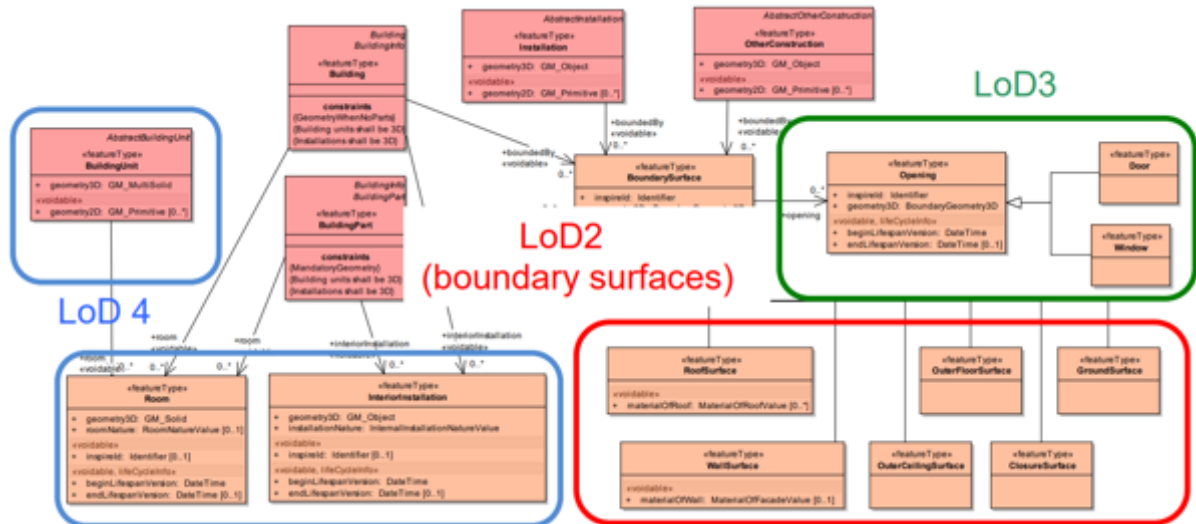


Figure 12 - CityGML LoD2, LoD3 and LoD4 in INSPIRE “Buildings” TG

6. Status of deployments

6.1 Indoor air quality and auxiliary data

According to Deliverable D4.3 (§6) the Activity plan foreseen in the period M13 – M24 provided the timeframe for deploying, configuring and testing the main software components enabling a minimum level of technical interoperability.

The following Table 29 refers to the general timeline planned in D4.3, with the status updated for each main component:

Table 29 General timeline for EDIAQI deployments

Software type	Software recommended	Month	13	14	15	16	17	18	19	20	21	22	23	24
Base software	- PostgreSQL - PostGIS		■	■	■	✓								
Software for dynamic data interoperability	- FROST				■	■	■	■	✓					
Software for interoperable view of web maps	- GeoServer							■	■	■	■	✓		
Software for interoperable access to IAQ indexes	- GeoServer							■	■	■	■	✗		
Pilot testing	- QGIS + STA plugin										■	■	■	✓

The following OGC SensorThings endpoints are available (access restricted with credentials):

- [Software for dynamic data - all IAQ data harvested from remote servers and concerning EDIAQI pilots and campaigns](#)
- [Software for dynamic data - IAQ data about Ferrara](#)
- [Software for dynamic data - IAQ data about Vilnius and Zagreb](#)
- [Software for interoperable view of web maps](#)
- [Know Centre Data Monitoring dashboard](#)

✗ Concerning “Software for interoperable access to IAQ” indexes, this will be postponed to M25 – M36 and linked to the pilots’ long-term objectives (T4.2); the idea is to implement in one or more pilots an Indoor Air Quality Index (IAQI) to describe the air quality in simple terms and an easy-to-understand colour scale.

In Table 30, a detailed plan for semantic interoperability was presented; hereafter, the table contains the updated status of each activity:



Table 30 Detailed plan for semantic interoperability

Action	Partners involved	Month	13	14	15	16	17	18	19	20	21	22	23	24
Complete physical mapping between EDIAQI dynamic data from pilots towards SensorThings data model	All T4.3 partners: DEDA (coord), TalTech, THIN, LAS, KNOW, WINGS, ASC, LC		■	■	✓									
Updated version of “data glossary” (chapter 3) with details about: - indoor biological parameters - indoor toxicological parameters - lifestyle metadata from questionnaires	All T4.3 partners			■	✓									
In-depth analysis of existing official vocabularies and ontologies	All T4.3 partners			■	■	✓								
Mapping towards CityGML data model (or INSPIRE “Buildings” data model)	All T4.3 partners			■	■	■	■	✓						
Updated version D4.3(v1.5)	All T4.3 partners							■	■	✓				
Analysis of standards for metadata (FAIR) and software solutions for discovery capabilities	All T4.3 partners								■	■	■	✓		
Specific T4.3 web meetings	All T4.3 partners			■		■		■		■		■		✓
Deliverable D4.7	DEDA										■	■	■	✓

In the following chapter **Error! Reference source not found..**

These implementations have been funded by DG JRC of European Commission within the API4INSPIRE project:

<https://github.com/DataCoveEU/API4INSPIRE>



Annexes, further details are provided for the following activities:

- Mapping towards CityGML data model.
- Analysis of standards for metadata (FAIR) and software solutions for discovery capabilities.



This project has received funding from the European Union's Horizon Europe Framework Programme under grant agreement N° 101057497.

The following table shows the different types of monitoring devices used in EDIAQI pilots and campaigns, with the number of devices in each pilot area and for the whole project, with list of parameters measured:

Table 31 Types of IAQ monitoring devices (with numbers and parameters)

Type of IAQ monitoring device	Ferrara #stations	Seville #stations	Estonia #stations	Vilnius #stations	Zagreb #stations	Overall #stations	#Observed Properties	Observed Properties
NetPID	32			5	3	40	10	PM10, PM2.5, CO, CO2, O3, NO2, TVOC, T, RH, PRESSURE
NetPID with AVOC	4					4	11	PM10, PM2.5, CO, CO2, O3, NO2, TVOC, AVOC, T, RH, PRESSURE
Awair		11				11	5	PM2.5, CO2, TVOCs, T, RH
Aeroqual		1				1	3	O3, NO2, Formaldehyde
Aranet PRO		5				5	4	CO2, T, RH, PRESSURE
Thinnect airPurity			102			102	9	PM10, PM2.5, PM1, CO2, O3, NO2, TVOC, T, RH
uRAD				2		2	10	PM10, PM2.5, PM1, CO2, O3, VOC, T, H, Pressure, Noise
AIRWINGS indoor/outdoor Vilnius				14		14	14	PM10, PM2.5, PM1, CO, CO2, NO2, T, RH, PRESSURE, NOISE, C2H5OH, CH4, H2, NH3
AIRWINGS indoor Zagreb					15	15	7	PM2.5, CO, CO2, TVOC, T, RH, PRESSURE
AIRWINGS outdoor Zagreb					10	10	13	CO, CO2, O3, NO2, NO, SO2, PM1, PM25, PM10, T, RH, PRESSURE, NOISE
IMROH indoor 1					5	5	12	PM1, PAHS (Fluoranthene, Pyrene, Benzo[a]anthracene, Chrysene, Benzo[j]fluoranthene, Benzo[b]fluoranthene, Benzo[k]fluoranthene, Benzo[a]pyrene, Dibenzo[a, h]anthracene, Benzo(ghi)perylene, Indeno[1, 2, 3-cd]pyrene)



IMROH indoor 2					5	5	13	PM1, Microplastics, PAHS (Fluoranthene, Pyrene, Benzo[a]anthracene, Chrysene, Benzo[j]fluoranthene, Benzo[b]fluoranthene, Benzo[k]fluoranthene, Benzo[a]pyrene, Dibenzo[a, h]anthracene, Benzo(ghi)perylene, Indeno[1, 2, 3-cd]pyrene)
IMROH outdoor					5	5	12	PM1, PAHS (Fluoranthene, Pyrene, Benzo[a]anthracene, Chrysene, Benzo[j]fluoranthene, Benzo[b]fluoranthene, Benzo[k]fluoranthene, Benzo[a]pyrene, Dibenzo[a, h]anthracene, Benzo(ghi)perylene, Indeno[1, 2, 3-cd]pyrene)
IMROH SKC Pocket Pump for VOCs						0	17	2-methylpentane, Methylene chloride, Methylcyclopentane, Chloroform, 2-methylhexane, Cyclohexane, Benzene, Heptane, Trichlorethylene, Methylcyclohexane, Toluene, tetrachlorethylene, Ethylbenzene, m-p-o xylene, Styrene, 1, 3, 5-trimethylbenzene, 1, 4-dichlorobenzene
IMROH Radon shortterm					70	70	1	Radon
IMROH Radon longterm					70	70	1	Radon
Xiaomi					12	12	1	Openings
Tellur					6	6	1	MotionDetection

Radiello passive sampler VOC	28					28	17	Methylcyclopentane, Cyclohexane, 2-Methylpentane, Heptane, 2-Methylhexane, Methylenechloride, Chloroform, Trichloroethylene, Tetrachloroethylene, 1, 4-dichlorobenzene, Benzene, Toluene, Ethylbenzene, o-xylene, p-xylene, Styrene, 1, 3, 5-trimethylbenzene
Radiello passive sampler Aldehydes							9	Formaldehyde, Acetaldehyde, Acrylaldehyde, Propionaldehyde, Butyraldehyde, Pentanal, Isovaleraldehyde, Glutaral, Hexanal, Benzaldehyde,
Plastic passive sampler				3		3	1	Microplastics
tot per pilot	64	17	102	24	213	420		

In addition to the 28 Radiello passive samplers VOC analysed in Ferrara, the following were positioned:

- 11 - Ferrara - for the autumn campaign not analysed due to overexposure
- 8 - Seville - lost in shipping
- 4 – Vilnius - lost in shipping
- 12 - Tallin - shipped on 11th November 2024

Radiello passive sampler Aldehydes will be positioned in 2025.

6.1 Outdoor air quality data

Concerning outdoor, air quality data are already available all over Europe through the same SensorThings API standard protocol.

These data are collected by the European Environmental Agency in the following server:

<https://discomap.eea.europa.eu/map/fme/AirQualityExport.htm>

The SensorThings endpoint is provided by the Fraunhofer Institute using FROST open-source solution and contains the complete time-series from January 2018 onwards:

<https://airquality-frost.k8s.ilt-dmz.iosb.fraunhofer.de/v1.1/>

Data can be viewed/browsed with the following web application:

<https://api4inspire.k8s.ilt-dmz.iosb.fraunhofer.de/servlet/is/127/>

These implementations have been funded by DG JRC of European Commission within the API4INSPIRE project:

<https://github.com/DataCoveEU/API4INSPIRE>



- **BUILDING:** in this table, spatial (3D) data and attribute of the whole building are stored and related each other and with other elements of other tables; main attributes of this table are the 3D geometry (referring to the SOLID describing the buildings manufact), number of storeys above and below ground, height, functions and uses, as well as the 2D footprint and the geometries of the intersection with terrain
- **ROOM:** in this table, the 3D geometry (SOLID) of each single room, with specific use and function are stored; this table is joined with both BUILDING (belonging to) and with SURFACE elements (exterior and interior walls)
- **SURFACE:** in this table, the 3D geometry of each exterior surface of BUILDING and interior surface of ROOM are stored
- **OPENING:** in this table, spatial 3D data about openings element existing on each surface are stored

In the context of EDIAQI, the needed information has been already collected in the LOCATIONS and in the FEATUREOFINTEREST classes of the SensortThings data model and can be easily mapped towards the CityGML schema.

The first case test can be the high school “Aleotti” located in Ferrara, with spatial data about rooms equipped with digital and analogical monitoring units that may be derived based on indoor maps already available (example below):



Figure 14 - High school "Aleotti" (Ferrara, Italy) indoor map at first floor



7.2 Standards for metadata (FAIR) and solutions for discovery capabilities

The European Commission requires and promotes data from granted projects to be FAIR⁷¹:

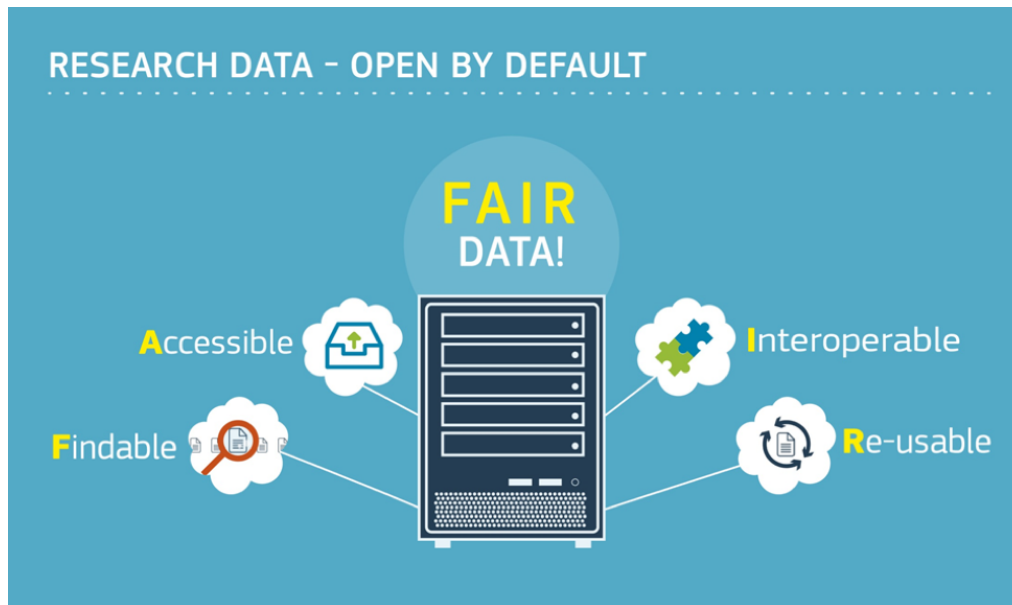


Figure 15 - FAIR data and EC requirements

Making EDIAQI data FAIR means to have them:

- **Findable** (i.e. discoverable with metadata): to make data reusable, first they have to be discoverable by means of clear and detailed metadata (data about data) understandable by both humans and computers (machine-readable metadata are essential for the discovery and the interoperability). Four sub-principles are to be considered:
 - F1. (Meta)data are assigned a globally unique and persistent identifier
 - F2. Data are described with rich metadata (defined by R1 below)
 - F3. Metadata clearly and explicitly include the identifier of the data they describe
 - F4. (Meta)data are registered or indexed in a searchable resource
- **Accessible** (i.e. always available and obtainable): once discovered, users need to know how to access them, including possible requirements for authentication and authorization (even if the data is restricted, the metadata must be open). The following sub-principles are to be considered:

⁷¹ <https://www.openaire.eu/how-to-make-your-data-fair>

- A1. (Meta)data are retrievable by their identifier using a standardised communication protocol
 - A1.1 The protocol is open, free, and universally implementable
 - A1.2 The protocol allows for an authentication and authorisation procedure, where necessary
- A2. Metadata are accessible, even when the data are no longer available
- **Interoperable** (i.e. both syntactically and semantically understandable): when accessing data, users need to understand them and may need to have them integrated with other data provided from other sources, using different software applications and workflows; this allows data exchange and reuse between researchers, institutions and organisations. The following sub-principles are to be considered:
 - I1. (Meta)data use a formal, accessible, shared, and broadly applicable language for knowledge representation.
 - I2. (Meta)data use vocabularies that follow FAIR principles
 - I3. (Meta)data include qualified references to other (meta)data
- **Reusable** (i.e. sufficiently described and shared with the least restrictive licences): the last principle is focused on optimizing the reuse of data, by providing all detailed information in the corresponding metadata, in particular IPR restrictions and use limitations, so to allow the widest reuse possible and the least cumbersome integration with other data sources. The following sub-principles are to be considered:
 - R1. (Meta)data are richly described with a plurality of accurate and relevant attributes
 - R1.1. (Meta)data are released with a clear and accessible data usage license
 - R1.2. (Meta)data are associated with detailed provenance
 - R1.3. (Meta)data meet domain-relevant community standards

Based on the FAIR principles and sub-principles, EDIAQI IAQ data should be described in (standard-based) metadata forms and made them findable through a “discovery service”.



A “discovery service” is a service making it possible to search for datasets on the basis of the content of the corresponding metadata and to display the content of the metadata⁷².

Indeed, many standards do exist to describe the content of a dataset with a metadata form; each of them may be peculiar of specific domains⁷³.

Two standards are suggested for metadata about EDIAQI IAQ data:

Standard for metadata (different domains)

EAD	Archiving	MARC	Librarianship
DDI	Archiving and Social Science	METS	Librarianship
CDWA	Arts	MODS	Librarianship
VRA Core	Arts	XOBIS	Librarianship
Darwin Core	Biology	RAD	Librarianship and archiving
ONIX	Book industry	PBCore	Media
CWM	Data warehousing	MPEG-7	Multimedia
EML	Ecology	MEI	Music notation
IEEE LOM	Education	DOI	Networked resources
MDDL	Financial market	Dublin Core	Networked resources
FOAF	Friend of a Friend (FOAF)	ISO/IEC 11179	Organizations
CSDGM	Geographic data	ISO 23081	Records management
ISO 19115	Geographic data	MoReq2010	Records management
e-GIMS	Government	DIF	Scientific data sets
GILS	Government/ Organizations	SAML	Software Systems
TEI	Humanities, social sciences and linguistics	ISO/IEC 19506	Software Systems
NISO MIX	Images	EBU CCDM	The EBU Class Conceptual Data Model – CCDM
NIEM	Law enforcement; Social services; Enterprise resource planning	EBUCore	The EBUCore metadata set for audio-visual content
		DCAT / DCAT-AP	Web resources, open data
		RDF	Web resources, linked data

Figure 16 - Standards for metadata in different domains

ISO19115 is the standard for geospatial data, required at EU level by the INSPIRE Directive (and its Regulations and Technical Guidelines)⁷⁴. In the context of EDIAQI, IAQ data are geolocated as they are related to specific “locations” (as described in chapter 0 and in particular in section 5.1.2 Locations). ISO19115 metadata have a machine-readable format expressed by XML encoding, as defined by ISO/TS19139.

⁷² <https://inspire.ec.europa.eu/glossary/DiscoveryService:1>

⁷³ <https://www.dcc.ac.uk/guidance/standards/metadata/list>

⁷⁴ https://knowledge-base.inspire.ec.europa.eu/publications/inspire-metadata-implementing-rules-technical-guidelines-based-en-iso-19115-and-en-iso-19119_en



DCAT-AP is the Application profile for data portals in Europe (DCAT-AP)⁷⁵ is a specification based on the W3C Data Catalogue vocabulary (DCAT)⁷⁶ an RDF⁷⁷ vocabulary designed to facilitate interoperability between data catalogues published on the Web. For DCAT-AP conformance, it is not mandatory that this happens in a RDF serialisation, but the exchanged format *SHOULD* be unambiguously transformable into RDF. JSON is a popular format to exchange data between systems, and DCAT-AP provides a JSON-LD context file⁷⁸. JSON-LD is a W3C Recommendation⁷⁹ that provided a standard approach to interpret JSON structures as RDF.

The two standards (ISO19115 and DCAT-AP) are mapped each other and to other existing metadata standard vocabularies (e.g. ISO 15836 Dublin Core, SMDX, ...). In particular, the GeoDCAT-AP extension⁸⁰ of the DCAT-AP has been released thanks to the joint initiative undertaken by several DGs of the European Commission in the context of the Interoperable Europe Programme⁸¹.

From an operational and pragmatical perspective, EDIAQI IAQ data may be FAIR by:

- make them **findable** with metadata based on **DCAT-AP standard specification**
- make **metadata accessible** through open, free, and universally implementable communication protocol, like the **open APIs provided by CKAN** open-source solution⁸²
- make data **accessible** and **interoperable** thanks to standard vocabularies (like **EIONET** for air pollutants and **ECHA** for chemical components) and open standard APIs for automatic data interaction, such as the **OGC SensorThings API**

⁷⁵ <https://op.europa.eu/en/web/eu-vocabularies/dcat-ap>

⁷⁶ <https://www.w3.org/TR/vocab-dcat/>

⁷⁷ <https://www.w3.org/RDF/>

⁷⁸ <https://semiceu.github.io/DCAT-AP/releases/3.0.0/context/dcat-ap.jsonld>

⁷⁹ <https://www.w3.org/TR/json-ld11/>

⁸⁰ <https://semiceu.github.io/GeoDCAT-AP/>

⁸¹ <https://joinup.ec.europa.eu/interoperable-europe>

⁸² <https://ckan.org/> and <https://extensions.ckan.org/extension/dcat/> and <https://docs.ckan.org/en/2.10/api/>



- make data **reusable** by enriching metadata with a clear, accessible and machine-readable open license (such as **the Creative Commons By Attribution**⁸³) and details about provenance and lineage of data as well as mechanisms to meet other domain-relevant community standards

It is noteworthy that the same standard (DCAT-AP) and software solution (CKAN) has also been identified by the [DG JRC web Data Catalogue](#).

In this catalogue, [examples of IAQ data](#) are available.

⁸³ <https://creativecommons.org/licenses/by/4.0/deed.en>





Deliverable D4.7

Framework and standards for data interoperability - version 2

Work Package 4

Pilots, data and campaigns

Version: Final



This project has received funding from the European Union's Horizon Europe Framework Programme under grant agreement N° 101057497.