



Connecting Europe Facility 2014-2020



AQMO

Air Quality and MObility

Grant Agreement Number: INEA/CEF/ICT/A2017/1566962

2017-FR-IA-0176

D7.3

AQMO Platform User Manual

Final

Version: 1.0

Author(s): François Bodin (UR1), Manuel Chevé (AirBreizh), Philippe Collinet (IDRIS), Benjamin Depardon (UCit), Simon Leray (AirBreizh), Laurent Morin (UR1), Olivier Oldrini (AmpliSIM)

Date: 22/12/2020

Project and Deliverable Information Sheet

| | | |
|---------------------------------------|---|---|
| AQMO Project | Project Ref. №: INEA/CEF/ICT/A2017/1566962 | |
| | Project Title: Air Quality and mobility | |
| | Project Web Site: http://aqmo.irisa.fr/ | |
| | Deliverable ID: <D7.3> | |
| | Dissemination Level: PU * | Contractual Date of Delivery: 31 / 12 / 2020 |
| | | Actual Date of Delivery: 22 / 12 / 2020 |
| EC Project Officer: Mark VELLA MUSKAT | | |
| Authorship | Written by: | François Bodin (UR1) |
| | Contributors: | Manuel Chevé (Air Breizh) Philippe Collinet (IDRIS) Benjamin Depardon (UCiT) Simon Leray (Air Breizh) Laurent Morin (UR1) Olivier Oldrini (AmpliSIM) |
| | Reviewed by: | Corentin Lefèvre (Néovia Innovation) |
| | Approved by: | TB/MB |

* - The dissemination level are indicated as follows: **PU** – Public, **CO** – Confidential, only for members of the consortium (including the Commission Services) **CL** – Classified, as referred to in Commission Decision 2991/844/EC.



Document Status Sheet

| Version | Date | Status | Comments |
|---------|------------|-------------------|---|
| 0.1 | 31/10/2020 | Draft version | Document structure Contribution partner UR1 general and from |
| 1.0 | 04/12/2020 | Augmented version | Contributions from other partners AirBreizh, AmpliSIM and IDRIS |
| 1.1 | 22/12/2020 | Final version | |

References and Applicable Documents

- [1] <https://www.seadatanet.org/>
- [2] <https://data.aqmo.org/>

List of Acronyms and Abbreviations

| | |
|-------|-------------------------------------|
| API | Application Programming Interface |
| AQMO | Air Quality and MObility |
| GPS | Global Positioning System |
| IT | Information Technology |
| JSON | Java Script Object Notation |
| LoRa | Long Range |
| MQTT | Message Queuing Telemetry Transport |
| NUC | Next Unit of Computing |
| OSM | Open Street Map |
| SenML | Sensor Measurement List |

Table of contents

| | |
|--|-----------|
| References and Applicable Documents..... | 4 |
| Introduction | 7 |
| 1 Adding a fix or mobile particle matter sensors..... | 8 |
| 1.1 Installing the sensors..... | 8 |
| 1.2 Installing the Central Unit..... | 10 |
| 1.3 Connection with the AQMO platform | 12 |
| 2 Monitoring and accessing the sensors' data | 13 |
| 2.1 Real time bus positioning and sensor values | 13 |
| 2.2 Sensor data access API..... | 14 |
| 2.1.1 Data schemes | 14 |
| 2.1.2 Applicable standards and norms | 15 |
| 2.1.3 API fonctionnalités | 16 |
| 3 Configuring and launching a simulation | 17 |
| 3.1 Definition simulation scenarios..... | 17 |
| 3.2 Simulation results | 21 |
| Conclusion | 25 |
| Annex A: Description of AQMO API services..... | 26 |
| Annex B: Data Schemes | 28 |

Table of figures

| | |
|--|----|
| Figure 1: Mobile sensor box (size L: 200mm, l: 165mm, h: 120mm)..... | 9 |
| Figure 2: Fix version of the sensor box..... | 9 |
| Figure 3: Positioning of the sensor box..... | 10 |
| Figure 4: Current version of the central unit..... | 10 |
| Figure 5: Bus side trunk..... | 11 |
| Figure 6: Central unit inside the bus side trunk..... | 11 |
| Figure 7: Installing the central unit in the side trunk of a bus..... | 12 |
| Figure 8: Real time visualization | 13 |
| Figure 9: Grafana dashboard..... | 14 |
| Figure 10: Bounding Box ("bbox") in GeoJson..... | 15 |
| Figure 11: Dashboard that lists all cases available for the current user logged in. | 18 |

| | |
|---|----|
| Figure 12: View of the calculation domain and the selection of the period to model, in this case it is the 4th January 2019. | 18 |
| Figure 13: Meteorological data on Rennes Metropolis area for 2019 and used for the simulations. The meteorological station is the yellow dot located in the airport area, and the data are presented as a wind rose. | 19 |
| Figure 14: View of the specie editor and the specie database | 20 |
| Figure 15: View of the point and line sources used for modeling the Rennes Metropolis area. | 20 |
| Figure 16: Map of average concentration of particulate matter during a specific day of 2019 in the Rennes Metropolis area. On the right-hand side, details about the data and the background maps available to the user. Concentrations of particulate matter are ranging from lower (in blue) to higher (in red). | 22 |
| Figure 17: Details on the impact of the street network close to Rennes city centre. Concentrations of particle matter are ranging from lower (in blue) to higher (in red). | 22 |
| Figure 18: Time slider used to browse through the various time frames. | 22 |
| Figure 19: View at different time of the day of the concentration of particulate matter. From top to bottom: 6 am, 8 am, 11 am and 2 pm, on the same day used for the daily average above. | 23 |
| Figure 20: Choice of a location for the pointer and extraction by the service of the time evolution on this specific location. | 24 |

Introduction

The AQMO platform provides three main functionalities that shall be understood by its users.

These functionalities are at the interfaces between the platform's results and its users (e.g. apart the platform's automation capabilities). The interfaces are physical (1. installing sensors) or virtual (2. monitoring and accessing the sensors' data ; 3. running a simulation).

This document aims to describe these three user level functionalities and how to interact with them.

Among the users of the platform, this document more specifically targets professional users who do not usually work in computing or air quality monitoring services but that use them or that can be concerned by the implementation of a mobile air quality monitoring system such as AQMO. To name a few, it could concern, for instance: bus maintenance workers (to install the mobile sensors) or specialist in urban planning (to use the simulation service). Other users may also be interested by the content of this report to acquire knowledge on how the platform is designed and how it delivers its results.

In this report, the AQMO team gives both a return of experience from the project but also gives details on the technical elements that were implemented to create the AQMO platform. By following this double approach, our aim is to give practical information on the platform's interfaces and how they concretely work, so that external users may understand how to use them. We have tried to give, as much as possible, illustrations of the work carried out during the project.

1 Adding a fix or mobile particle matter sensors

On one end, the AQMO platform rely on a network of air quality monitoring “stations” that comprise elements that, altogether, form the AQMO edge computing capability. Those elements are:

- An air quality sensor¹
- A communication module
- A GPS
- A NUC
- Communication and power connectors

All those elements are put in a box that can be installed at a fixed point² or in mobility – in our case on buses of the Rennes Metropolis transport network.

A dedicated package has been designed and tested by University Rennes 1 with the support of Keolis and AirBreizh to build the AQMO mobile sensor box that integrates all the elements needed by the AQMO's specifications, both on the edge computing side and the physical bus constraints. 3-D printing has been used and allowed the design to be adapted to the environment of the station.

A specific focus has been put to monitor the effort required to install such boxes on a bus as a critical element for replicability.

To set up the measurement and the data upload devices there are two main operations:

- 1) Installing the sensors
- 2) Installing the central unit to setup the local storage, computation and the communication links

The installation process takes about 7hours (4h for installing the power lines, drilling the hole in the bus body, 3h for the sensors and the central unit) for one bus.

The two operations are described in the next two sections.

1.1 Installing the sensors

Installing the sensors can be performed in two factor forms. The first one for mobile use showed in Figure 1 and the second one for a fix position (Figure 2). The mobile sensor box is glued on top of the front bus body as shown in Figure 3.

¹ More details can be read in D2.1, D2.2 and D6.2

² Fig. 2 shows such an installation done at the University Rennes 1 premises that was used for various testing purposes

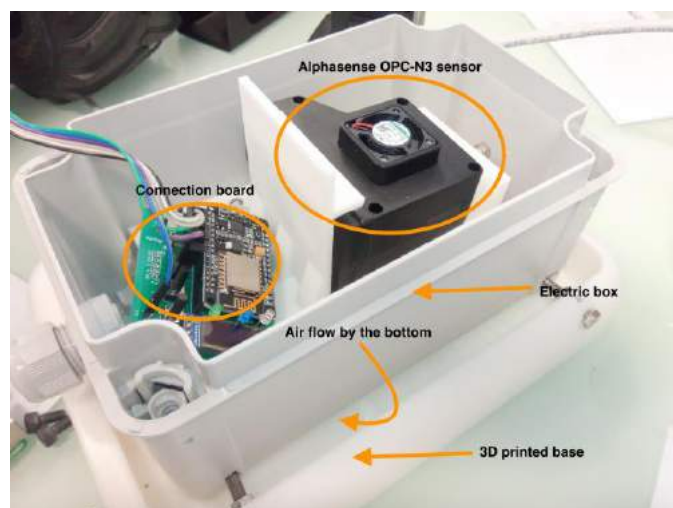


Figure 1: Mobile sensor box (size L: 200mm, l: 165mm, h: 120mm).



Figure 2: Fix version of the sensor box installed at the University Rennes 1 Campus



Figure 3: Position of the sensor box on a bus

1.2 Installing the Central Unit

Prior to the installation of the central unit, two power lines have to be installed:

- 1) One going from the bus electrical panel to the central unit (24V);
- 2) One power line from the central unit to the sensor box (5V). This requires two make a hole in the bus body. The central unit does provide the 5V power line. A junction box (not shown) is set on the top of the bus to allow for supplying more sensors.

Figure 4 shows the current version of the central unit (size is L: 310mm, I: 250mm, h: 150mm). Figure 5 & Figure 6 & Figure 7 show the setup of the central unit inside a bus.



Figure 4: Current version of the central unit.



Figure 5: Bus side trunk

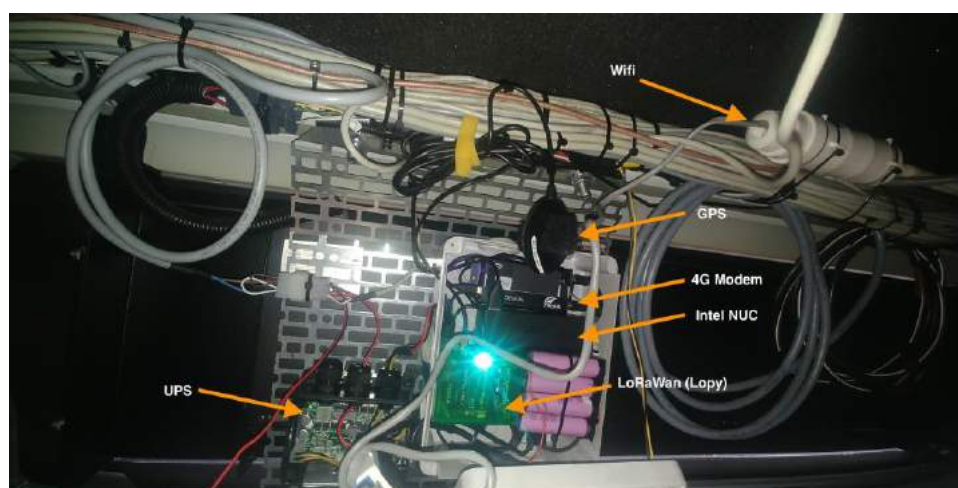


Figure 6: Central unit inside the bus side trunk



Figure 7: Installing the central unit in the side trunk of a bus

1.3 Connection with the AQMO platform

Once a new station has been installed, whether on a fixed venue or in mobility, it shall be connected to the AQMO's network of sensors so that it can exchange its data to fuel the AQMO platform.

In the AQMO's project, this connection is done with a "Primary Server" that has been created by the University of Rennes 1.

This step is not performed by the users themselves as it requires special administration rights to integrate a new sensor to the network, both for technical and security reasons.

2 Monitoring and accessing the sensors' data

One of the main purpose of the AQMO platform for its users is to access the data acquired through data visualization systems. The data are displayed through two complementary means that can be used for different purposes.

On one side visualization tools (map and dashboard) are created for the users to be able to access at any time an overlook of the data gathered. Real-time bus positioning has proven to be an important feature for monitoring purposes while sensor's values help air quality professionals to take an overlook of the evolution of the air quality situation.

On the other side and for deeper integration into other systems or exploitation, a dedicated API has been created to access the data.

These two functionalities are described in the following sections.

2.1 Real time bus positioning and sensor values

Real-time bus positioning and sensors values can be accessed simply through a web browser interface. This “easy-to-access” way allows any kind of user to be able to read, check and use AQMO data visualization as it required no technical background.

Figure 8 shows the real-time map with the sensors positions and sensors values. The position of buses can be approximated (pink marker) because of the lack of synchronization between the GPS position acquisition and the particle matter measurement. Figure 9 shows the Grafana dashboard reporting the sensor values.³

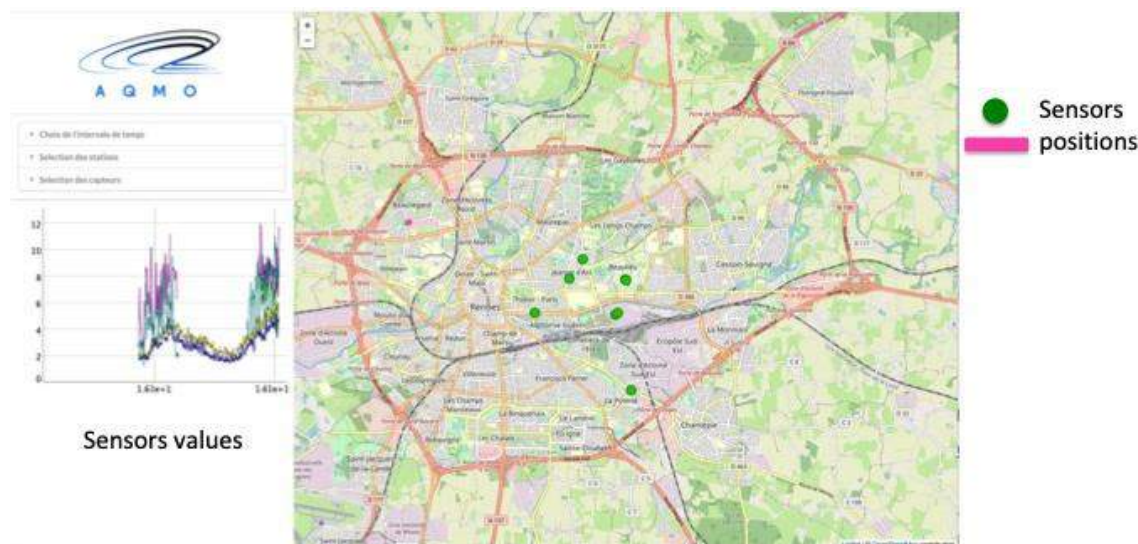


Figure 8: Real time visualization interface

³ See also D5.3 “Report on the deployment of visualization tools for the AQMO platform”

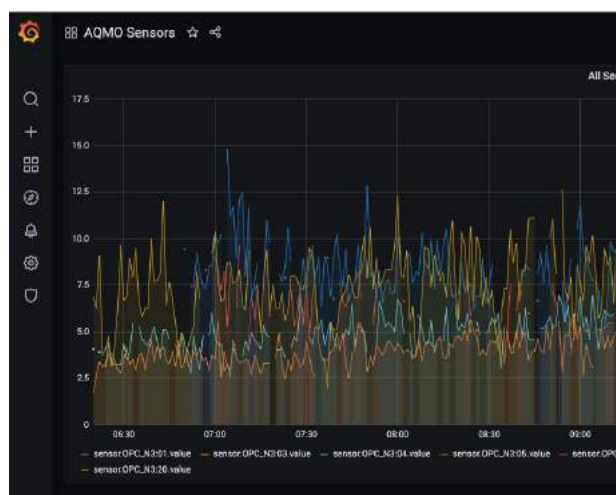


Figure 9: Grafana dashboard

2.2 Sensor data access API

The purpose of the data access interface is to provide the data sets obtained in real time. An API is nowadays a common system but requires that the user has an access to professional or at least sufficient IT skills. This section provides insight on the data format used, the application standards and norms and finally the access API.

2.1.1 Data schemes

Data schemes have been designed to satisfy fundamental properties in the capture, provision, qualification, and cycle management of this data. These properties, simple and short, are as follows:

- All data is structurally positioned in space and time. There is no constraint on the level of precision of positioning but it is mandatory.
- Data are produced in time series. They must necessarily refer to an object at the source of the data: the operational producer of the data. The latter must be named in a structured and global way.
- The data producer, the name/type of the measurement made and the unit used for its capture must be specified. These elements are positioned and interpreted in the producer's repository.
- Data producers can be fixed or mobile. The notion of measurement station or station designates the technical platform in charge of supporting the data producer (shelters, power supply, interconnection, sensors). The spatio-temporal positioning of the data can be completed by a reference to the station in charge of carrying out the measurement. It is important to note that a station can encompass multiple sensors.
- The quality and qualification of the data must be monitored throughout the life of the data. In this respect, two properties are natively integrated within the data series: the qualitative status of the data (according to the SeaDataNet [1] reference system) and the data cycle management status.

The data schemes are given in Annex B.

2.1.2 Applicable standards and norms

The norms and standards that have been chosen for the implementation of the data schemes are : the data format is Json as standardized in RFC 8259. The Json Format is used end-to-end on the mobile air quality capture platform through various transmission systems (LoRa, MQTT, etc.). The geographic data format is the GeoJson as standardized in RFC 7946 .

The GeoJson format is the root measurement storage and exchange format for the AQMO platform. The data are positioned in space by geographical coordinates (bbox, see Figure 10) and preferably with points delimiting the point tolerance. For specific and geolocalised queries, results can be rendered in the form of sub-documents (properties, data series).

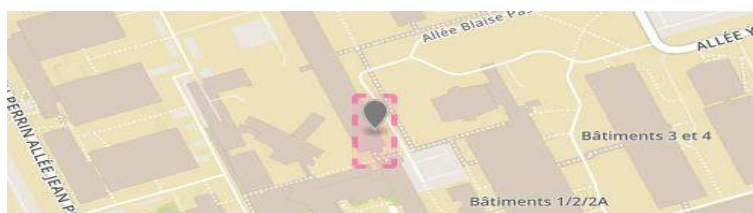


Figure 10: Bounding Box ("bbox") in GeoJson

Temporal positioning is integrated in two ways: by the station and by the measurement series itself. The following principles have been used to represent the data:

- The sensor data format is the Sensor Measurement Lists (SenML) as standardised in RFC 8428. This sensor representation format is given in Annex B. A GeoJson entry may contain multiple SenML sections, with complete or partial SemML data sets. The names and units of measurement used should conform to current standards where possible.
- The measurement data, their properties, and the data series are in principle non-mutable (or read-only): once produced, they cannot be modified. This implies that any new production of derived data generates a new data series. In the event of a change of property, it must be possible to add a deprecation flag to the source data, and the origin of the data must be specified for the derived data. As an exception, two properties can be added to SenML data:
 - o The new state of the data in the implementation of the data cycle. The property must integrate a sequence number.
 - o A deprecation property pointing to the derived data replacing the produced data.
- The qualitative representation format for sensor data uses the SeaDataNet SDN:L20 standard. This qualitative data representation format allows the status of the data to be formally represented: raw, correct, unlikely, invalidated, etc. The SDN:L20 property is introduced as an extension to the SenML format for each data produced.
- The representation format of the "data cycle" as defined in the principle section is not yet specified.
- The format for the representation of "stations" as defined in the principle section is not standardized at this stage. Stations are optionally embedded as a section of the GeoJson Object Property List. They must include the standard name of a measurement station, the start and end dates, and optionally the policy for interpreting the positioning data - fixed station, mobile, type of movement, etc. - as well as the data type.

2.1.3 API fonctionnalités

The access service for the mobile pollution sensor platform is provided at the following address: <https://data.aqmo.org/>. [2]

The API provides the following services:

- Access to station list
- Access to sensor list
- Access to sensors data and bus positions.

They are detailed in Annex A.

3 Configuring and launching a simulation

The third main user interface of the AQMO platform is related to simulation capacities that target mainly professional users, especially in the air quality monitoring and urban planning areas.

The simulation interface of the project is based on the tools developed by AmpliSIM. The simulation service is based on the data gathered by the platform that are completed with a range of other data as described below.

The simulation service is designed for both simple and more complex manipulations. Simple versions are based on running predefined scenarios (pre-selection of simulation parameters) that are updated with data when a user decides to run a computation. Using scenarios does not require specific skills and so can be used by any user. A more complex version is also available that allows the user to fine-tune its simulation parameters and thus create scenarios. This targets experienced users. As the simple version is extremely easy to use once the user is connected to the interface, only the definition of simulation scenarios is described in this document.

In both cases simulation results are displayed through a graphical interface described in this section. It shall be noticed that simulation results can be made accessible to a broader audience than professional users (e.g. citizen, decision-makers...).

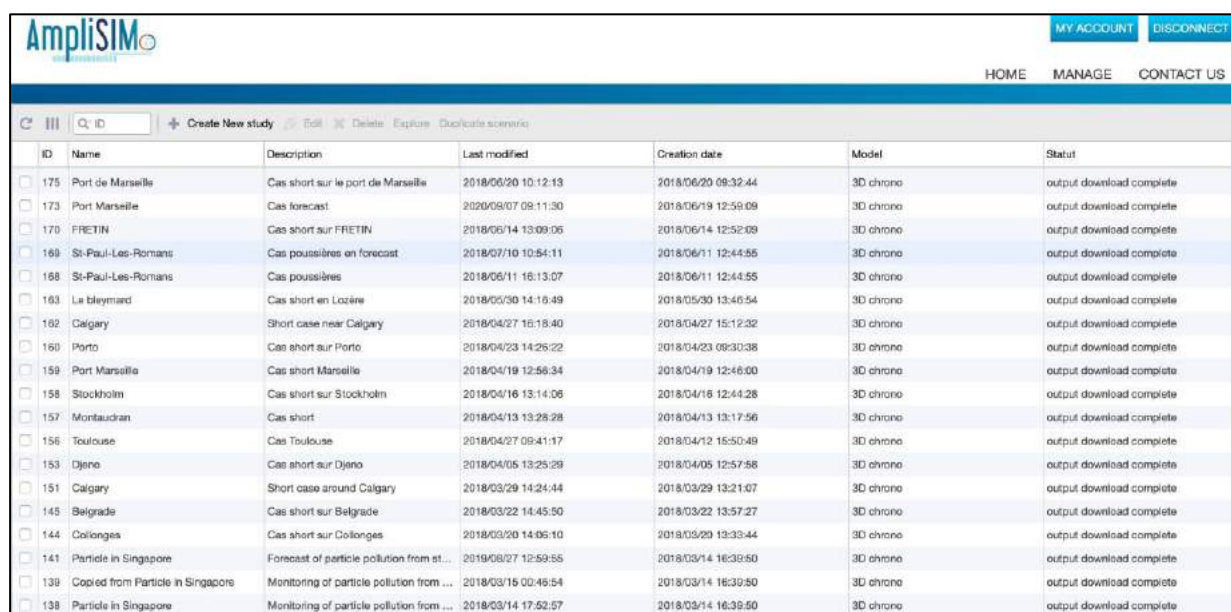
3.1 Definition simulation scenarios

The expert user wants to define scenarios to compare their impact on the air quality. Exploration of the results as described in the previous section is obviously important. But at his level the capability to define precisely the scenarios is equally important.

This section hence describes the way the user can define multiple scenarios.

Through the dashboard (see Figure 11), the user can easily manipulate various scenarios related to the same study. He can:

- List them, and browse through older scenarios,
- Share them with other users,
- Create new scenarios from scratch,
- Duplicate them to create alternate scenarios starting from the same configuration: the complete setup of the original scenario is hence available from the start. The user can focus on modifying only the different hypothesis related to that scenarios



| ID | Name | Description | Last modified | Creation date | Model | Status |
|-----|----------------------------------|---|---------------------|---------------------|-----------|--------------------------|
| 175 | Port de Marseille | Cas short sur le port de Marseille | 2018/06/20 10:12:13 | 2018/06/20 09:32:44 | 3D chrono | output download complete |
| 173 | Port Marseille | Cas forecast | 2020/05/07 09:11:30 | 2018/06/19 12:59:09 | 3D chrono | output download complete |
| 170 | FRETIN | Cas short sur FRETIN | 2018/06/14 13:09:06 | 2018/06/14 12:52:09 | 3D chrono | output download complete |
| 169 | St-Paul-Les-Romans | Cas poussières en forecast | 2018/07/10 10:54:11 | 2018/06/11 12:44:55 | 3D chrono | output download complete |
| 168 | St-Paul-Les-Romans | Cas poussières | 2018/06/11 16:13:07 | 2018/06/11 12:44:55 | 3D chrono | output download complete |
| 163 | La bleyard | Cas short en Lozère | 2018/05/30 14:16:49 | 2018/05/30 13:46:54 | 3D chrono | output download complete |
| 162 | Calgary | Short case near Calgary | 2018/04/27 16:18:40 | 2018/04/27 15:12:32 | 3D chrono | output download complete |
| 160 | Porto | Cas short sur Porto | 2018/04/23 14:26:22 | 2018/04/23 09:30:38 | 3D chrono | output download complete |
| 159 | Port Marseille | Cas short Marseille | 2018/04/19 12:58:34 | 2018/04/19 12:46:00 | 3D chrono | output download complete |
| 158 | Stockholm | Cas short sur Stockholm | 2018/04/16 13:14:06 | 2018/04/16 12:44:28 | 3D chrono | output download complete |
| 157 | Montaudran | Cas short | 2018/04/13 13:28:28 | 2018/04/13 13:17:56 | 3D chrono | output download complete |
| 156 | Toulouse | Cas Toulouse | 2018/04/12 09:41:17 | 2018/04/12 15:50:49 | 3D chrono | output download complete |
| 153 | Djano | Cas short sur Djano | 2018/04/05 13:25:29 | 2018/04/05 12:57:58 | 3D chrono | output download complete |
| 151 | Calgary | Short case around Calgary | 2018/03/29 14:24:44 | 2018/03/29 13:21:07 | 3D chrono | output download complete |
| 145 | Belgrade | Cas short sur Belgrade | 2018/03/22 14:45:50 | 2018/03/22 13:57:27 | 3D chrono | output download complete |
| 144 | Colonges | Cas short sur Colonges | 2018/03/20 14:06:10 | 2018/03/20 13:33:44 | 3D chrono | output download complete |
| 141 | Partiele in Singapore | Forecast of particle pollution from st... | 2019/03/27 12:59:55 | 2018/03/14 16:39:50 | 3D chrono | output download complete |
| 139 | Coped from Particle in Singapore | Monitoring of particle pollution from ... | 2018/03/16 00:46:54 | 2018/03/14 16:39:50 | 3D chrono | output download complete |
| 138 | Partiele in Singapore | Monitoring of particle pollution from ... | 2018/03/14 17:52:57 | 2018/03/14 16:39:50 | 3D chrono | output download complete |

Figure 11: Dashboard that lists all cases available for the current user logged in.

When creating a scenario from scratch, the user follows, through the service, the process below:

- Defining the site and the time period,
- Defining the meteorological data,
- Setting up the emissions,
- Modifying expert parameters, if he requires,
- Defining the output.

The site creation is performed directly on the map (see Figure 12). The domain, where the calculation is performed, can be drawn on the map, and the time period is defined in the left panel. Geographical data, such as the simulation domain, can either be manipulated directly on the map but also in the left panel using the coordinates.

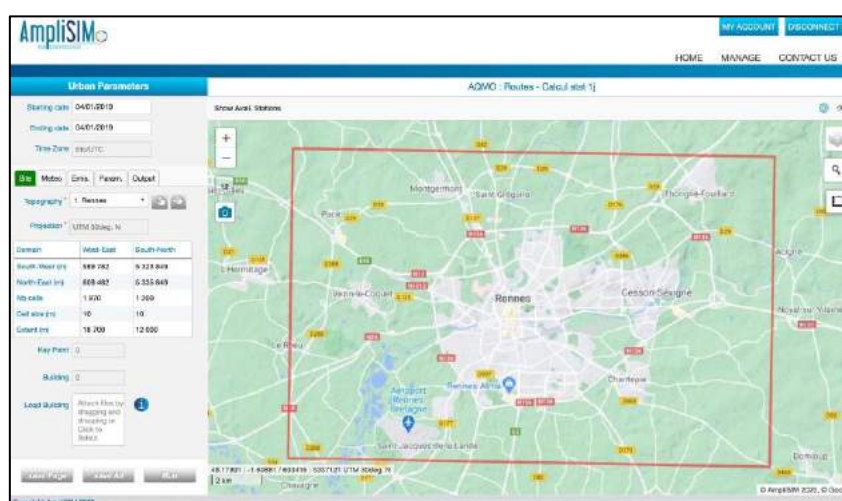


Figure 12: View of the calculation domain and the selection of the period to model, in this case it is the 4th January 2019.

Meteorology is defined in three ways:

- From available measurements stations,
- From meteorological mesoscale modeling, available for the past period (re-analysis), or in forecast (6/12/24/48/72h),
- From custom measurement uploaded by the user.

In the first case the user is choosing from the meteorological stations in the vicinity of the domain and provided by the service. In the second case, the data is automatically extracted for the period by the service for the location of the calculation. In the last case, the position of the meteorological station is defined on the map by the user and the associated data is dragged and dropped through a file in the service.

Figure 13 is presenting the wind rose for 2019 on the Rennes Metropolis area and obtained through the service from the meteorological station located at the Rennes airport. The wind rose can also be plotted on a monthly basis. Wind data, and the other meteorological data (temperature, pressure, humidity, cloud cover, ...) can be analysed too as time series.

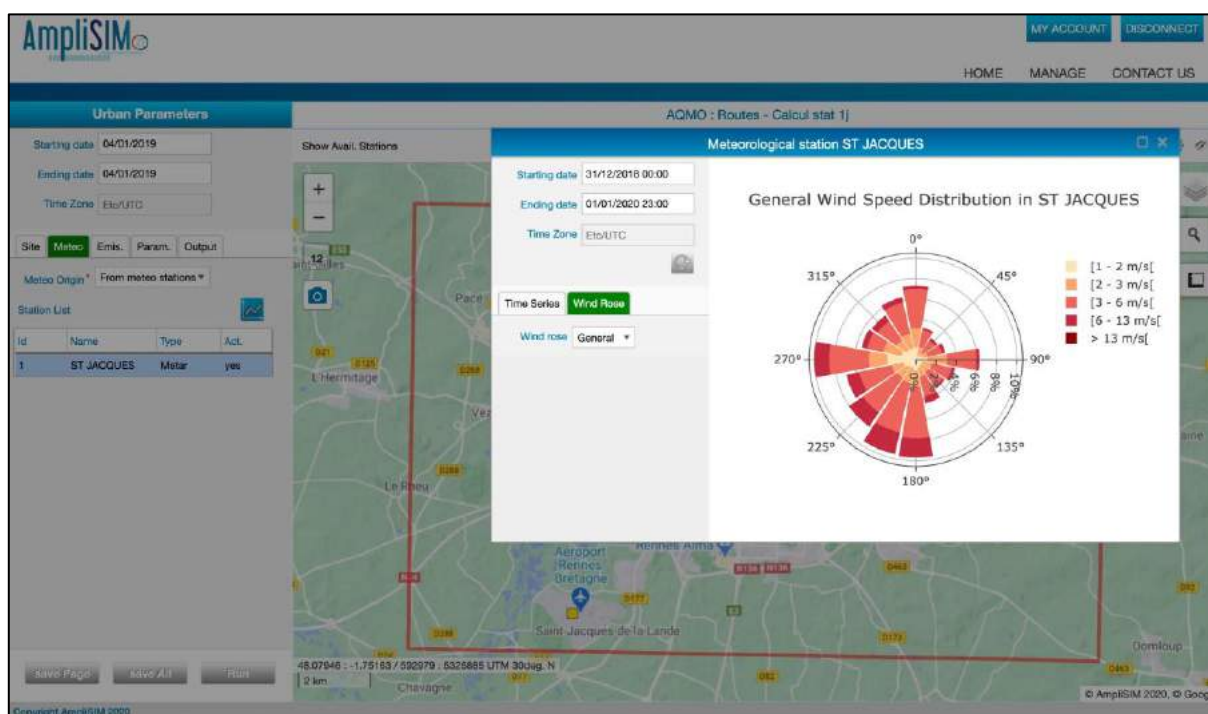


Figure 13: Meteorological data on Rennes Metropolis area for 2019 and used for the simulations. The meteorological station is the yellow dot located in the airport area, and the data are presented as a wind rose.

Manipulation of the emissions is more complex, and requires three components:

- The definition of the geometry of the various sources: point sources for stacks, area source for instance for agriculture, and line source usually but not exclusively related to road traffic emissions,
- The definition of the species to be released. The service is providing a complete database of pollutants along with their physical characteristics (molar mass, deposition speed, etc.), see Figure 14. Additional pollutants can be defined by the user.

- The actual definition of the emissions, i.e. the time evolution of the rate of emission for each specie emitted and for each source

The definition of the geometry of the sources can be draw on the map. Still, for a large number of sources such as line sources for a road network in a metropolis area, import of data is advised. A drag and drop of a geometry file is available, using the open specification and widely compatible shapefile format. Direct import from external API, such as Open Street Map (OSM) is also available.



Figure 14: View of the specie editor and the specie database

The source setup for Rennes is shown in Figure 15. It contains 10 stacks and more than 10,000 road sources.

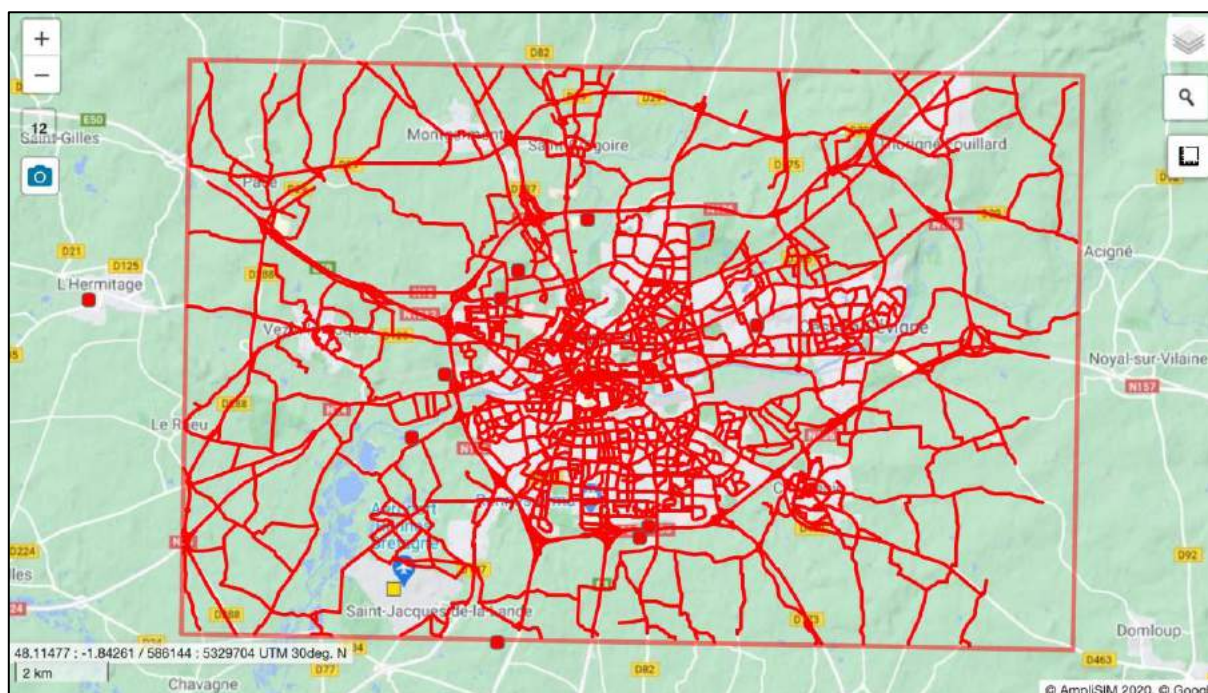


Figure 15: View of the point and line sources used for modeling the Rennes Metropolis area.

Regarding the actual definition of the emission, they can be defined in three ways:

- Constant emissions per source
- Modulation emissions: daily, weekly and monthly modulations are defined and combine to produce modulation profiles. To each and every source is attached one of these modulation profiles, and combined to the constant emission,
- Arbitrary emissions per source. These are to be specified at each hourly time step. They can be defined through the drag and drop of tabulated emission files.

The definition of specific parameter is associated closely to the model being used, and in case of urban modeling with NO_x calculation, often integrate parameters for the NO/NO₂ conversion.

The definition of output encompasses the choice of species available as output, together with the typology of statistics: averages, number of days above a specific threshold, centiles. It allows also asking to provide hourly data as map, but also to realize specific treatments like assessment mixing concentration values and population data to compute specific exposition.

Once this setup is defined, the user can launch the calculation. It can also, before, or after the launch, duplicate the setup to start an alternate scenario that can be deployed at the same time thanks to the elasticity of the computing infrastructure.

3.2 Simulation results

This section is dedicated to data that are accessible to the end-user, being the general public or a high-level decider. As such, it consists in practice in simulation results, i.e. mainly maps of data. Indeed, the end-user is searching for data that has direct impact on himself, like level of pollution or air quality.

Results are displayed, after being logged in, and selected through browsing with the dashboard.

For instance, Figure 16 presents an average daily concentration of PM10 computed using the SIRANE model for a specific day of 2019. The user can access various navigation options:

- Pan and zoom the map,
- Selection of the background map (plan, satellite, topography, etc.),
- Selection of the pollutants,
- Access values of pollutants on specific locations,
- Change the colour scale, export the data, etc.



Figure 16: Map of average concentration of particulate matter during a specific day of 2019 in the Rennes Metropolis area. On the right-hand side, details about the data and the background maps available to the user. Concentrations of particulate matter are ranging from lower (in blue) to higher (in red).

Thanks to the high resolution of the calculation, the user can zoom in and see the influence of the buildings and traffic emission at the street level. Figure 17 displays the level of details in the close vicinity of Rennes city centre.

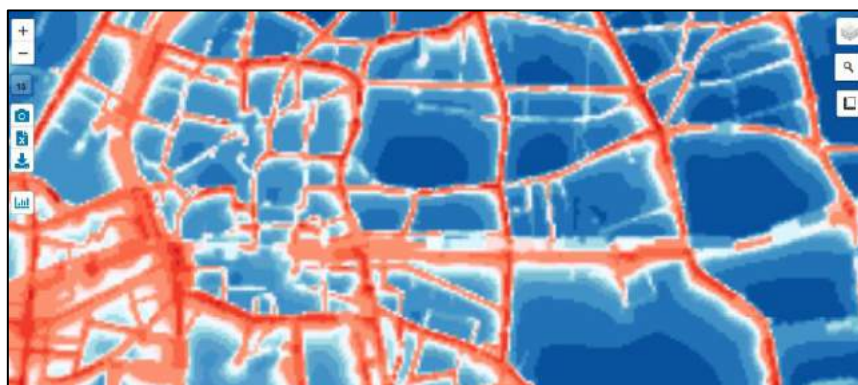


Figure 17: Details on the impact of the street network close to Rennes city centre. Concentrations of particle matter are ranging from lower (in blue) to higher (in red).

Data are not only available as averages over period of time (day/week/month/year), but time evolutions are also accessible. To manipulate time evolving data, a time slider is available (see Figure 18) that allows the user to browse through the various time frames.



Figure 18: Time slider used to browse through the various time frames.

This allows the user of the service to get a view of the evolution of air quality, due to the automotive traffic, but also the evolution of the meteorological conditions, throughout the day. The Figure 19 shows maps of particulate matter concentration for various moments of the same day chosen in the figures above where the daily average was presented. The influence of the commuting throughout the day, but also the wind direction, is obvious.

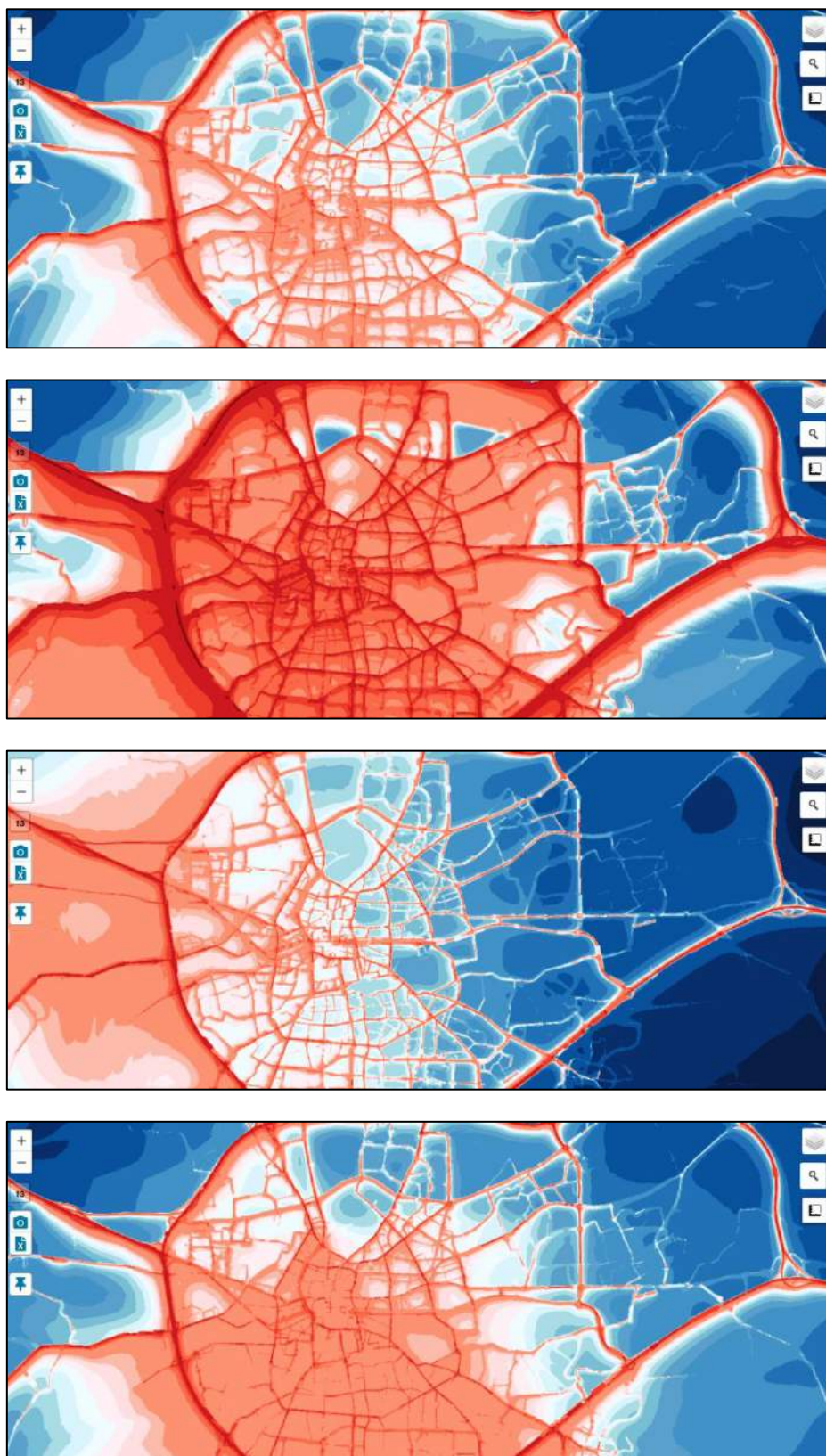


Figure 19: View at different time of the day of the concentration of particulate matter. From top to bottom: 6 am, 8 am, 11 am and 2 pm, on the same day used for the daily average above.

The time evolution of the concentration for the various contaminants can also be explored through the use of graphics pointing: the user simply drags and drops the pointer and get the precise evolution at some location he chose on the map. The process is illustrated on the Figure 20.

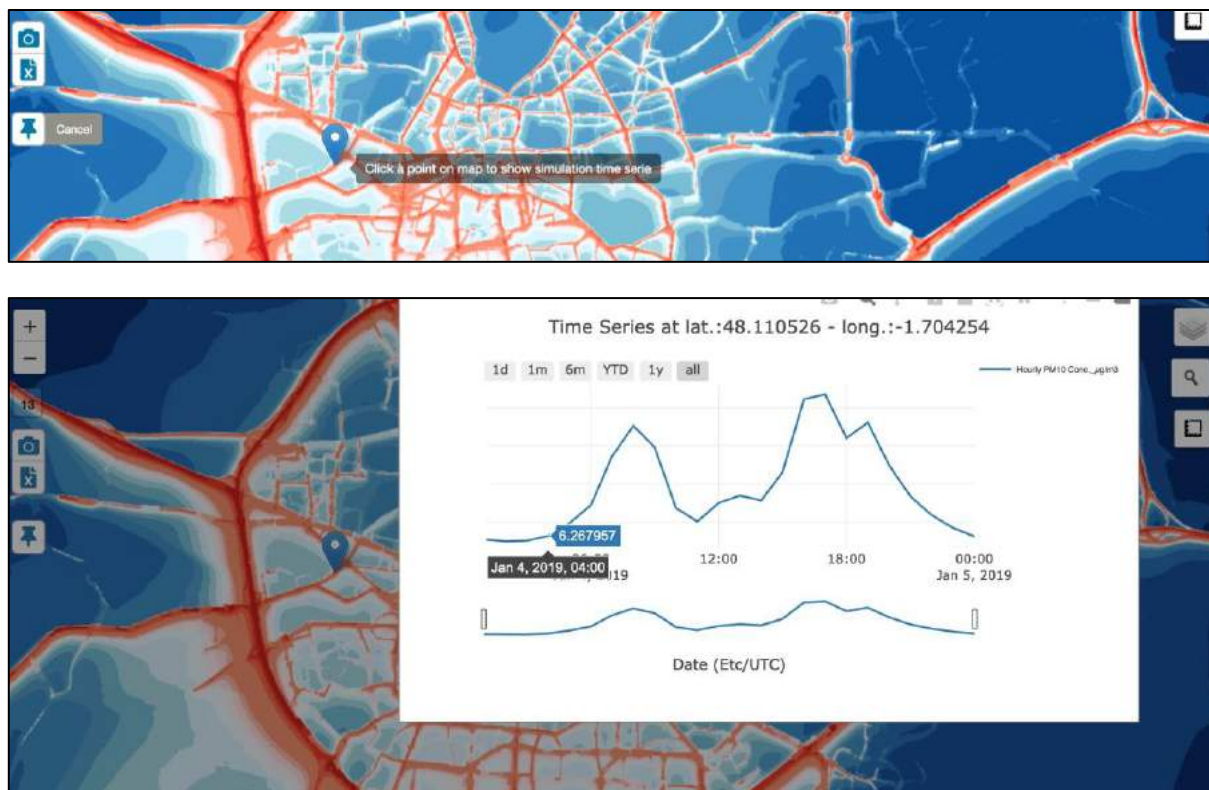


Figure 20: Choice of a location for the pointer and extraction by the service of the time evolution on this specific location.

Multiple scenarios can also be explored, since the simulation allows to model the actual chain of events, but also to explore hypothetical scenarios: change in the composition of the automotive fleet, change in the road planning, etc. The complete set of cases available can be listed through a dashboard (see Figure 11). This content is only accessible if the user is logged in, and the cases available are only simulation cases public to the user.

Conclusion

The AQMO platform benefits different types of users and shall then offer useable interfaces to fit different purposes. Visualization tools are at the heart of the data display – should these data be related to air quality monitoring or air quality simulation results.

AQMO offers a range of interactions that takes into account the needs and skills of different users and thus offers a variety of services that can address at the same time users as different as citizen, air quality professionals, urban planners, decision-makers...

The description of the physical interface related to the installation of new sensors will also help different kind of users, still it mainly describes the operations needed to install air quality monitoring stations in mobility, which constitutes one the project's main innovation.

Annex A: Description of AQMO API services

| | | |
|---|-----------------------------|------------------------|
| <i>GET</i> | api/sensors | List available sensors |
| List all available sensors found in the DB. Takes no parameters. | | |
| Response Class | | |
| [String] | | |
| Example | | |
| <pre>["LOPY-AQ03", "LOPY-AQ07", "OPC_N3:04", "OPC_N3:12", "atmoTrack:100200060:pm1", "atmoTrack:100200060:pm10", "atmoTrack:100200060:pm25", "atmoTrack:100200060:rhume", "atmoTrack:100200060:temp", "atmoTrack:100200061:pm1", "atmoTrack:100200061:pm10", "atmoTrack:100200061:pm25", "atmoTrack:100200061:rhume", "atmoTrack:100200061:temp", "atmoTrack:100200263:pm1", "atmoTrack:100200263:pm10", "atmoTrack:100200263:pm25", "atmoTrack:100200263:rhume", "atmoTrack:100200263:temp"]</pre> | | |

| | | |
|--|------------------------------|-------------------------|
| <i>GET</i> | api/stations | List available stations |
| List all available stations found in the DB. Takes no parameters. | | |
| Response Class | | |
| [String] | | |
| Example | | |
| <pre>["atmoTrack-100200060", "atmoTrack-100200061", "atmoTrack-100200263", "parautarin02", "standalone-LOPY-AQ03"]</pre> | | |

| | | | | |
|---|------------------------------|----------------------------|----------------|--|
| GET | api/measures | Return measures | | |
| <p>Request a list of measure from the DB. No parameters are required, but the response scope is changed depending on the parameters.</p> <p>Without parameters, the result is limited to a single entry. This results is for testing purpose.</p> <p>With the station specified, the result returns the last known position of the station or the position at the date specified if date boundaries are provided.</p> <p>With the sensor specified, the result returns the last known measure of the sensor.</p> <p>The previous sensor modes can be modified by the date field, but remains limited to a single entry.</p> <p>Location mode extends the limit in number of results to 100. Two modes are available :</p> <ul style="list-style-type: none">• The point mode with the fields “lat” and “lon” only. <p>This mode select measurements in a radius of 500 meters.</p> <ul style="list-style-type: none">• The region mode, the point mode extended with the fields “e_lat” and “e_lon” <p>This mode select measurements in a square made by the coordinates.</p> | | | | |
| Parameters | | | | |
| station | | The name of the station | String | |
| sensor | | The name of the sensor | String | |
| date | | The measurement date end | Epochtime (ms) | |
| from_date | | The measurement date start | Epochtime (ms) | |
| lat | | The latitude of search | Float | |
| lon | | The longitude of search | Float | |
| e_lat | | The maximum latitude | Float | |
| e_lon | | The maximum longitude | Float | |
| Response Class | | | | |
| GeoJSON | | | | |

Annex B: Data Schemes

The following appendices present the Json schematics as defined by the JSON Schema Draft 6 standard. The schemas have been put online at the reference addresses (<https://data.aqmo.org/schemas>) and can be used to validate API requests. They will be updated with the API and are always up to date. The root schema, recursively loading all schemas, can be referenced as follows:

```
{ $ref: "https://data.aqmo.org/schemas/geojson" }
```

GeoJson Data Scheme

```
{
  $schema: "http://json-schema.org/draft-07/schema",
  $id: "https://data.aqmo.org/schemas/geojson",
  title: "The Root GeoJson Schema", description: "Geo location",
  $defs: {
    point: {
      $id: "#/geojson/point", title: "Location Point",
      description: "Define a 2D or 3D GPS point",
      type: "array", default: [ 0, 0 ],
      items: [
        { type: "number", minimum: -180, maximum: 180},
        { type: "number", minimum: -180, maximum: 180},
        { type: "number", minimum: 0 }
      ],
      minItems: 2, maxItems: 3, additionalProperties: false
    },
    coords: {
      $id: "#/geojson/coords", title: "Coordinates",
      description: "Define a set of GPS points",
      type: "array",
      items: { $ref: "#/$defs/point" },
      additionalProperties: false
    },
    station: { $ref: "https://data.aqmo.org/schemas/station" },
    SenML: { $ref: "https://data.aqmo.org/schemas/SenML" },
    "SDN:L20": { $ref: "https://data.aqmo.org/schemas/SDN:L20" },
    DataCycle: { $ref: "https://data.aqmo.org/schemas/DataCycle" }
  },
  type: "array",
  items: {
    title: "Geo data ",
    type: "object", required: [ "geometry", "type" ],
    properties: {
      type: {
        $id: "#/geojson/type", title: "The GeoJson structure type",
        type: "string", enum: ["Feature", "FeatureList"],
        default: "Feature", additionalProperties: false
      },
      geometry: {
        $id: "#/geojson/geometry", title: "The Geometry Schema",
        type: "object", required: [ "type", "coordinates" ],
        properties: {
          type: {

```

```

    $id: "#/geojson/geometry/type", title: "Position type",
    type: "string", enum: ["LineString", "Point"],
    default: "Point", additionalProperties: false
  },
  coordinates: {
    $id: "#/geojson/geometry/coordinates", title: "GPS position",
    anyOf: [ { $ref: "#/$defs/point" }, { $ref: "#/$defs/coords" } ],
    additionalProperties: false
  }
},
additionalProperties: false
},
bbox: {
  $id: "#/geojson/bbox", title: "GPS bounding box",
  anyOf: [ { $ref: "#/$defs/point" }, { $ref: "#/$defs/coords" } ],
  additionalProperties: false
},
"properties": {
  $id: "#/geojson/properties", title: "Sensor Properties",
  type: "object", required: [ "station" ],
  patternProperties: {
    station: { $ref: "#/$defs/station" },
    ".*:.*": { $ref: "#/$defs/SenML" }
  },
  additionalProperties: false
}
}
}
}

```

Station Data Scheme

```

{
  $schema: "http://json-schema.org/draft-07/schema",

  $id: "https://data.aqmo.org/schemas/station",
  title: "Station Schema",
  description: "Define the state of a mobile or a fixed sensor station",
  $defs: {
    date: {
      $id: "#/date", title: "UTC Date",
      description: "Define an UTC date either using the POSIX time (Unix Epoch) or the 'YYYY-MM-DDTHH:mm:ss.sssZ' format",
      type: [ "integer", "string" ],
      additionalProperties: false
    },
  },

  type: "object", required: [ "name", "date", "radius", "policy" ],
  properties: {
    name: {
      $id: "#/station/name", title: "Station name",
      description: "The unique and univversal name of a mobile or a fixed sensor station",
      type: "string",
      additionalProperties: false
    },
  },
}

```

```

date: {
  $id: "#/station/date", title: "Station status date",
  description: "The date or the ending date of station state description",
  $ref: "#/$defs/date"
},
from_date: {
  $id: "#/station/from_date", title: "End of Station status date",
  description: "The ending date of station state description",
  $ref: "#/$defs/date"
},
radius: {
  $id: "#/station/radius", title: "Position radius (m)",
  description: "The position radius in meters, a circle defining the
positioning precision",
  type: "number",
  additionalProperties: false
},
policy: {
  $id: "#/station/policy", title: "The location evaluation policy",
  description: "The name of the policy used to compute the position of the
station",
  type: "string", enum: ["fixedGps", "mobileGps"],
  additionalProperties: false
}
},
additionalProperties: false
}

```

SDN:L20 Data Scheme

```

{
  $schema: "http://json-schema.org/draft-07/schema",
  $id: "https://data.aqmo.org/schemas/SDN:L20",
  title: "SeaDataNet measurand qualifier flags",
  description: "Flags used to provide additional information, usually
referring to data quality, about the data values",
  anyof: [ {
    type: "integer", minimum:0, maximum:11, // description: "SDN:L20
identifier value"
  }, {
    type: "string", // description: "SDN:L20 Identifier",
    enum: [ "SDN:L20::0", "SDN:L20::1", "SDN:L20::2", "SDN:L20::3",
"SDN:L20::4", "SDN:L20::5",
"SDN:L20::6", "SDN:L20::7", "SDN:L20::8", "SDN:L20::9", "SDN:L20::A",
"SDN:L20::Q" ]
  }, {
    type: "string", // description: "SDN:L20 Label",
    enum: [
      "none",    // No quality control procedures have been applied to the data
value. This is the initial status for all data values
entering the working archive.
      "good",    // Good quality data value that is not part of any identified
malfunction and has been verified as consistent with real
phenomena during the quality control process.
      "probably_good", // Data value that is probably consistent with real
phenomena but this is unconfirmed or data value forming

```

```

        part of a malfunction that is considered too small to
        affect the overall quality of the data object of which it
        is a part.
    "probably_bad", // Data value recognised as unusual during quality
        control that forms part of a feature that is probably
        inconsistent with real phenomena.
    "bad", // An obviously erroneous data value.
    "changed", // Data value adjusted during quality control. Best practice
        strongly recommends that the value before the change be
        preserved in the data or its accompanying metadata.
    "BD", // The level of the measured phenomenon was less than the limit
        of detection (LoD) for the method employed to measure it.
        The accompanying value is the detection limit for the
        technique or zero if that value is unknown.
    "excess", // The level of the measured phenomenon was too large to be
        quantified by the technique employed to measure it. The
        accompanying value is the measurement limit for the
        technique.
    "interpolated", // This value has been derived by interpolation from
        other values in the data object.
    "missing", // The data value is missing. There should be no
        accompanying value in ODV format files. The accompanying
        value in SeaDataNet NetCDF data must be the absent data
        representation specified by the _FillValue parameter
        attribute and lie outside the range of data not flagged bad
        (4) or probably bad (3).
    "ID_uncertain", // There is uncertainty in the description of the
        measured phenomenon associated with the value such as
        chemical species or biological entity.
    "BelowLOQ" // The level of the measured phenomenon was less than the
        limit of quantification (LoQ). The accompanying value is
        the limit of quantification for the analytical method or
        zero if that value is unknown.
    ]
    }]
}

```

DataCycle Data Scheme

```

{
  $schema: "http://json-schema.org/draft-07/schema",
  $id: "https://data.aqmo.org/schemas/DataCycle",
  title: "Sensor Measurement Data Cycle",
  description: "Represent the policy and the status of the data in the data-
cycle management",
  anyof: [{
    type: "integer", minimum:0, maximum:11, // Abstract representation of the
status // Temporary
    additionalProperties: false
  }, {
    type: "string", // Domain, Policy, and Data Cycle status,
    pattern: "^[a-zA-Z_]+:[a-zA-Z_]+:[a-zA-Z_]+$",
    additionalProperties: false
  }]
}

```

SenML Data Scheme

```
{
  $schema: "http://json-schema.org/draft-07/schema",
  $id: "https://data.aqmo.org/schemas/SenML",
  title: "Sensor Measurement Lists",
  description: "Define the measurement time serie (records) of a simple
               sensor with the associate device parameters",
  $defs: {
    date: {
      $id: "#/date", title: "UTC Date",
      description: "Define an UTC date either using the POSIX time (Unix
                   Epoch) or the 'YYYY-MM-DDTHH:mm:ss.sssZ' format",
      type: [ "integer", "string" ],
      additionalProperties: false
    },
    "SDN:L20": { $ref: "https://data.aqmo.org/schemas/SDN:L20" },
    DataCycle: { $ref: "https://data.aqmo.org/schemas/DataCycle" }
  },

  type: "array",
  items: {
    $id: "#/SenML/fields", title: "Sensor Measurement fields",
    description: "Fields available in a SenML record",
    type: "object",
    properties: {
      bn: {
        $id: "#/SenML/bn", title: "Base Sensor Name",
        description: "Base Name of the sensor, global and universal if unique,
                     or when prepended to the following sensor names",
        type: "string", additionalProperties: false
      },
      bt: {
        $id: "#/SenML/bt", title: "Base Time",
        description: "Base and absolute time of the sensor, added to the
                     following sensor time",
        $ref: "#/$defs/date", additionalProperties: false
      },
      bu: {
        $id: "#/SenML/bu", title: "Base Unit",
        description: "Base Unit, see the RFC 8428, and IETF initiative for the
                     list",
        type: "string", additionalProperties: false
      },
      bv: {
        $id: "#/SenML/bv", title: "Base Value",
        description: "Base value added to the following measurement values",
        type: "number", additionalProperties: false
      },
      bs: {
        $id: "#/SenML/bs", title: "Base Sum",
        description: "Base value of the integrated sum of the values over time",
        type: "number", additionalProperties: false
      },
      bver: {
        $id: "#/SenML/bver", title: "Base Version",
```



```

    description:"Version value of the SemML entry. The value defines the
        minimal level of functionalities required for the
        entry",
    type:"integer", additionalProperties:false
},
n:{
    $id:"#/SenML/n", title:"Sensor Name",
    description:"Name of the sensor, global and universal if prepended by
        the base sensor name defined before",
    type:"string", additionalProperties:false
},
u:{
    $id:"#/SenML/u", title:"Unit",
    description:"Measurement Unit, see the RFC 8428, and IETF initiative
        for the list",
    type:"string", additionalProperties:false
},
v:{
    $id:"#/SenML/v", title:"Value",
    description:"Value of the entry using a floating point format",
    type:"number", additionalProperties:false
},
vs:{
    $id:"#/SenML/vs", title:"String Value",
    description:"String value of the entry",
    type:"string", additionalProperties:false
},
vb:{
    $id:"#/SenML/vb", title:"Boolean Value",
    description:"Boolean Value of the entry",
    type:"boolean", additionalProperties:false
},
vd:{
    $id:"#/SenML/vd", title:"Data Value",
    description:"Binary representation of the entry value using the base64-
        url encoding format",
    type:"string", additionalProperties:false
},
s:{
    $id:"#/SenML/s", title:"Integrated Sum",
    description:"Integrated sum of the values over time in floating point
        format",
    type:"number", additionalProperties:false
},
t:{
    $id:"#/SenML/t", title:"Measurement Time",
    description:"Date/time of the value. Relative if a base time exists,
        absolute otherwise",
    anyof: [ { $ref: "#/$defs/date" }, "number" ],
    additionalProperties:false
},
ut:{
    $id:"#/SenML/ut", title:"Update Time",
    description:"Period in seconds, represents the maximum time before this
        sensor will provide an updated reading for a
        measurement",
    type:"number", additionalProperties:false
},
bp:{

```

```

    $id:"#/SenML/bp", title:"Base time precision",
    description:"Base precision added to the following precision values",
    type:"number", additionalProperties:false
  },
  p:{
    $id:"#/SenML/p", title:"Time precision in seconds",
    description:"Time precision of the measurement, in seconds",
    type:"number", additionalProperties:false
  },
  dc:{
    $id:"#/SenML/dc", title:"Data-cycle policy and status",
    description:"Represent the policy and the status of the data in the
                  data-cycle management",
    $ref: "#/$defs/DataCycle", additionalProperties:false
  },
  "SDN:L20":{
    $id:"#/SenML/bp", title:"Measurement Status",
    description:"Status of the measurement using the SDN:L20 referential",
    $ref: "#/$defs/SDN:L20", additionalProperties:false
  }
},
additionalProperties: false
}

```