



## **Connecting Europe Facility 2014-2020**



**AQMO**

**Air Quality and MObility**

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

**2017-FR-IA-0176**

**D3.2**

**Documentation of the AQMO extended simulation services**

***Final***

Version: 1.0

Author(s): Benjamin Depardon (UCiT), François Bodin (UR1), Manuel Chevé (Air Breizh), Olivier Oldrini (AmpliSIM), Philippe Collinet (IDRIS), Simon Leray (Air Breizh).

Date: 09/12/2020

## Project and Deliverable Information Sheet

<b>AQMO Project</b>	<b>Project Ref. №:</b> INEA/CEF/ICT/A2017/1566962	
	<b>Project Title:</b> Air Quality and mobility	
	<b>Project Web Site:</b> <a href="http://aqmo.irisa.fr/">http://aqmo.irisa.fr/</a>	
	<b>Deliverable ID:</b> <D3.2>	
	<b>Dissemination Level:</b> PU *	<b>Contractual Date of Delivery:</b> 31 / 10 / 2020
		<b>Actual Date of Delivery:</b> 09 / 12 / 2020
<b>EC Project Officer:</b> Mark VELLA MUSKAT		
<b>Authorship</b>	<b>Written by:</b>	François Bodin (UR1)
	<b>Contributors:</b>	Benjamin Depardon (UCiT) Manuel Chevé (Air Breizh) Olivier Oldrini (AmpliSIM) Philippe Collinet (IDRIS) Simon Leray (Air Breizh)
	<b>Reviewed by:</b>	Corentin Lefèvre (Néovia Innovation)
	<b>Approved by:</b>	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 V1	
0.2	04/12/2020	Reviewed version	
1.0	09/12/2020	Final version	

## References and Applicable Documents

*List all external documents referenced in this document*

- [1] COPERT V: methodology for calculating pollutant emissions from road traffic set up within the AASQA network and based on the evolution of emissions as a function of vehicle speed ("The industry standard emissions calculator"; <https://www.emisia.com/utilities/copert/>).
- [2] CITEPA : Inter-Professional Technical Centre for Air Pollution Studies
- [3] CLC : Corine Land Cover, inventaire biophysique de l'occupation des sols et de son évolution selon une nomenclature en 44 postes
- [4] ICPE : Installation Classée pour la Protection de l'Environnement
- [5] The Delta Tool has been developed by the Joint Research Centre, the European Commission's science and knowledge service.
- [6] MAJIC: population distribution data based on INSEE, IGN and DGfip data
- [7] <https://www.airbreizh.asso.fr/publication/saint-brieuc-armor-agglom>
- [8] <https://www.airbreizh.asso.fr/publication/brest-metropole-evaluation-de-la-qualite-de-lair/>
- [9] <https://plotly.com/javascript/>
- [10] Soulhac, L., Salizzoni, P., Cierco, F.-X. et Perkins, R. J., 2011. The model SIRANE for atmospheric urban pollutant dispersion: PART I: presentation of the model. Atmospheric Environment. Volume 45, Issue 39, 7379-7395, (<http://air.ec-lyon.fr/SIRANE/>)

## List of Acronyms and Abbreviations

*Below is an extensive the List of Acronyms used in previous deliverables. Please add additional ones specific to this deliverable and delete unrelated ones.*

2D	Two Dimension
3D	Three Dimension
AASQA	Association Agréée de Surveillance de la Qualité de l'Air – Certified Association for Air Quality Monitoring
ACB	AmpliSIM Connection Backend
AD	Active Directory
AI	Artificial Intelligence
AIR	Atmosphere Impact and Risk

AMWS	AmpliSIM Modelling Web Service
API	Application Programming Interface
ARPEGE	<i>Action de Recherche Petite Echelle Grande Echelle</i> – Small Scale Large Scale Research Action
AQMO	Air Quality and MObility
AWS	Amazon Web Service
CCME	Cloud Cluster Made Easy
CITEPA	<i>Centre Interprofessionnel Technique d'Etudes de la Pollution Atmosphérique</i> – Interprofessional Technical Center for Atmospheric Pollution Studies
CLC	CORINE Land Cover
CLI	Command-Line Interface
CNRS Research	<i>Centre National de la Recherche Scientifique</i> – National Centre for Scientific Research
COPERT	Computer Program to calculate Emissions from Road Transport
CPU	Central Processing Unit
CU	Central Unit
DARI	<i>Demande d'Attribution de Ressources Informatiques</i> – Computing Resources Allocation Request
DEISA	Distributed European Infrastructure for Supercomputing Applications
EBS	Elastic Block Store
EC	European Commission
EC2	Elastic Cloud Computing
ECL	<i>Ecole Centrale de Lyon</i> – Lyon Central Engineering School
EFA	Elastic Fabric Adapter
EIP	Elastic Internet Protocol
EU	European Union
FGC	<i>Formulaire de Gestion de Compte</i> – Account Management Form
GB	Giga Byte
GENCI	<i>Grand Equipement National de Calcul Intensif</i> – Big National Equipment Intensive Computing

GPFS	General Parallel File System
GPU	Graphic Processing Unit
HPC	High Performance Computing
IAM	Identity and Access Management
ICPE	<i>Installation Classée pour la Protection de l'Environnement</i> – Classified Facility for the Protection of the Environment
IDRIS	Institute for Development and Resources in Intensive Scientific Computing
IGN Information	<i>Institut Géographique Nationale</i> – National Institute of Geographic and Forest
IP	Internet Protocol
IT	Information Technology
JSON	JavaScript Object Notation
KMS	Key Management Service
LMFA	Laboratory on Fluid Mechanics and Acoustics
MAJIC	<i>Mise à Jour des Installations Cadastreales</i> – Update of Land Register Facilities
MPI	Message Passing Interface
OS	Operating System
PM	Particulate Matter
PPST	<i>Protection du Potentiel Scientifique et Technique de la Nation</i> – Protection of the National Scientific and Technical Potential
RAM	Random Access Memory
S3	Simple Storage Service
SCP	Service Control Policy
SG	Security Group
SLURM	Simple Linux Utility for Resource Management
SOAP	Simple Object Access Protocol
SSH	Secure Shell
TB	Tera Byte
TFA	Two Factor Authentication

TTL	Time To Live
UTC	Universal Time Coordinated
URL	Uniform Resource Locator
VLAN	Virtual Local Area Network
VM	Virtual Machine
VPC	Virtual Private Cloud
VPN	Virtual Private Network
WMS	Workflow Management System

## Table of contents

<b>Introduction .....</b>	<b>12</b>
<b>1 Extended service description overview.....</b>	<b>12</b>
<b>2 Sirane numerical simulation.....</b>	<b>13</b>
<b>3 Urban Model Methodology and Validation .....</b>	<b>15</b>
3.1.1 <i>Creation of an optimised network of interconnected streets .....</i>	15
3.1.2 <i>Calculation of pollutant emissions - related to road traffic .....</i>	16
3.1.3 <i>Calculation of pollutant emissions related to other sources.....</i>	17
3.1.4 <i>Consideration of weather conditions .....</i>	17
3.1.5 <i>Determination of background pollution.....</i>	18
<b>3.2 Representation of the studied territory and validation of the model .....</b>	<b>18</b>
<b>3.3 Review of the methodology and its validation .....</b>	<b>19</b>
3.3.1 <i>Constraints on calculation performance vs. output precision.....</i>	19
<b>4 IDRIS – AQMO set up architecture and security constraints .....</b>	<b>20</b>
<b>4.1 Context and regular usage .....</b>	<b>20</b>
<b>4.2 Usage modalities .....</b>	<b>20</b>
<b>4.3 Account management: account opening and closure .....</b>	<b>21</b>
4.3.1 <i>Opening a user account .....</i>	21
4.3.2 <i>Closure of a user account .....</i>	22
4.3.3 <i>Access to the resources .....</i>	23
<b>4.4 IDRIS - AQMO security architecture and usage .....</b>	<b>24</b>
4.4.1 <i>IDRIS WEB service portal architecture .....</i>	24
4.4.2 <i>WEB service portal confinement .....</i>	24
4.4.3 <i>WEB service portal access.....</i>	25
4.4.4 <i>WEB service portal account rights: .....</i>	25
4.4.5 <i>WEB service to HPC links specifications: .....</i>	25
4.4.6 <i>WEB service access tracking.....</i>	26
<b>4.5 IDRIS – AQMO setup process .....</b>	<b>26</b>



4.5.1	WEB service portal setup process .....	26
4.5.2	Simulation applications to IDRIS infrastructure adjustment .....	26
4.5.3	IDRIS – AQMO setup Rex.....	26
4.5.4	IDRIS – AQMO cost model .....	26
4.6	HPC as a service access at IDRIS.....	28
5	AWS set up architecture and security constraints .....	28
5.1	Solution architecture overview .....	28
5.2	Cluster architecture.....	29
5.3	Security .....	32
5.4	HPC as a service access .....	33
6	AmpliSIM service access API.....	33
7	Data sensors integration and visualization .....	33
7.1	Integration .....	33
7.2	Visualization.....	34
8	Conclusion.....	34
	ANNEXE 1: AmpliSIM simulation service API .....	35

## List of Figures

Figure 1:	The part in the orange box implements the simulation as a service in the AQMO platform. ....	12
Figure 2:	Conceptual model of SIRANE (source: ECL - LMFA). ....	13
Figure 3:	Illustration of street-canyon volumes in SIRANE (source: ECL - LMFA). ....	14
Figure 4:	Process of creating an Air Breizh urban model; .....	15
Figure 5:	Emissions of nitrogen oxides NO <sub>x</sub> from the road network in Rennes Métropole. ....	16
Figure 6:	representation of 2016 annual PM10 emissions (residential sector). ....	17
Figure 7:	Illustration and meaning of an evaluation target "Target Plot" from Delta Tool. ....	18
Figure 8:	illustration and meaning of a Quantile-Quantile figure.....	19
Figure 9:	IDRIS HPC cluster architecture. ....	23

Figure 10: AQMO to IDRIS HPC architecture. ....	24
Figure 11: High-level view of the solution. ....	28
Figure 12: Cluster standard architecture.....	30
Figure 13: AWS services used to build a Cloud cluster.....	31
Figure 14: Plot of the time evolution of the NOx measurements at a sensor location.....	34

## Executive Summary

This report presents the approach made by AQMO to provide pollutant dispersion numerical simulation.

In a nutshell the main elements are:

- 1) The SIRANE model has been successfully integrated. SIRANE is a state of the art numerical model for pollutant dispersion
- 2) A deployment of the simulation as a service has been developed at IDRIS by AmpliSIM
- 3) The simulation as a service is also available on Cloud resources thanks to UCiT
- 4) A methodology for validating the simulation has been implemented by Air Breizh
- 5) Access to IDRIS computing resources has been adapted to allow simulation as a service
- 6) An API to easily create, run, configure simulations provided by AmpliSIM

One important conclusion is that AQMO results allows to envision the deployment of a service at national (EU level) for air quality related simulations that would serve not only the organisms in charge of air quality surveillance but also metropolis that would use this tool to study changes in transportation etc. (i.e. explore ‘what if’ scenario). Based on AQMO’s results as described in this report, such a service could be extended at a European level.

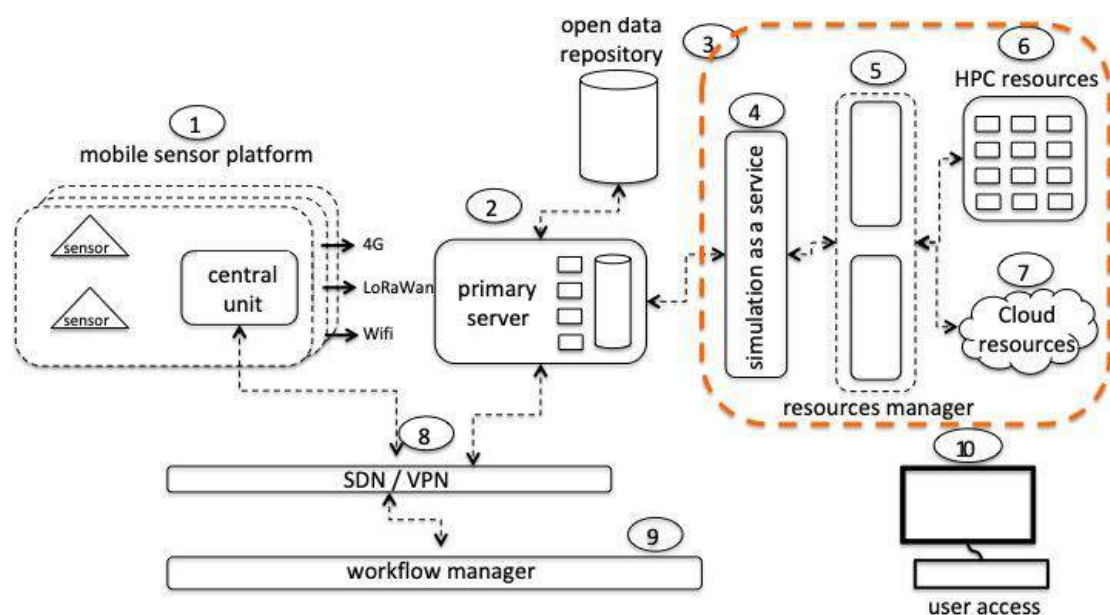
## Introduction

The air quality monitoring and its improvement in an urban area requires the full cooperation of local administrations, citizens, and industrial partners. The AQMO project objective is to provide an end-to-end platform able to collect the air pollution measurements and perform numerical simulation of pollutant dispersion.

This document describes the approach chosen for providing the simulation of pollutant dispersion as a service. Section 1 gives an overview of the AQMO architecture. Section 2 presents the SIRANE numerical model used. Section 3 described the methodology used to perform the simulations. Sections 4 and 5 deal with the computing resources allocation. Section 6 describes the main elements of the API accessing the simulation service. Section 7 addresses some of the data visualisation and integration aspects.

## 1 Extended service description overview

The Figure 1 recalls the architecture of the AQMO platform. The simulation as a service is bounded by the orange limit. In this box there are 4 main components (numbered 4, 5, 6, and 7). Component 4 corresponds to the API and service deployment. The fifth element is the two resources manager, one for the IDRIS (6) HPC center and the other one (7) for using Cloud resources (in our case AWS). The computing resources run the SIRANE model developed by the Ecole Centrale de Lyon (ECL).



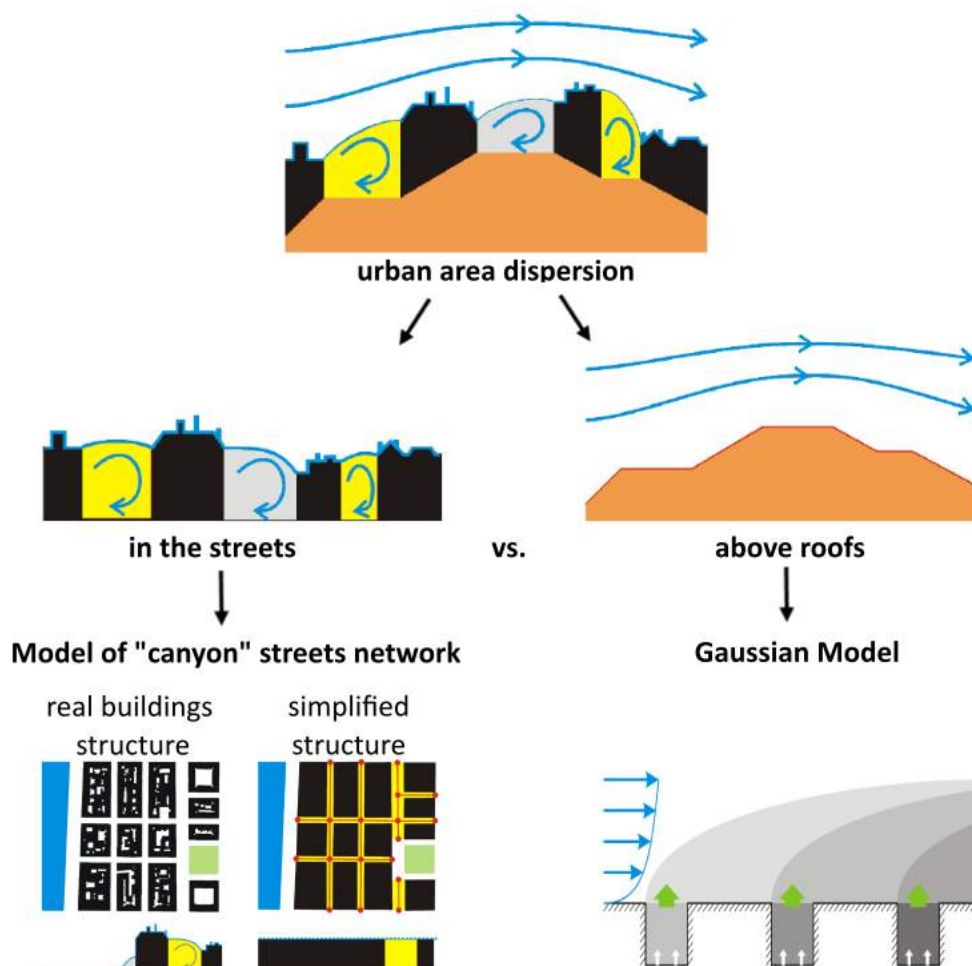
**Figure 1:** The part in the orange box implements the simulation as a service in the AQMO platform.

## 2 Sirane numerical simulation

The SIRANE model / software has been developed for 20 years by the AIR - Atmosphere, Impact & Risk team of the Laboratory of Fluid Mechanics and Acoustics (LMFA) of the Ecole Centrale de Lyon (ECL).

This model was created to simulate the dispersion of air pollutants at the scale of an agglomeration and at the level of its neighbourhoods. It describes the spatial and temporal (hourly) evolution of pollutant concentrations in an urbanised area in the form of maps or temporal evolution.

SIRANE is currently used by Air Breizh and other AASQA to model air quality in urban areas. It has been designed to take into account different situations from the simplest (an isolated point emission source) to the most complex urban modelling (multiple industrial sources associated with emissions from the residential/tertiary and road transport sectors in an urban area).



**Figure 2:** Conceptual model of SIRANE (source: ECL - LMFA).

The model distinguishes between two types of physical phenomena affecting the dispersion of pollutants in the urban environment (see Figure 2):

1. At street level, containment of pollutants between buildings ("canyon" street phenomena) and exchange of pollutants at street intersections;
2. Above the roofs, exchange between the streets and the "free atmosphere" and then dispersion of the Gaussian puff type.

Meteorological pre-treatment and atmospheric stability characterisation allows the vertical structure of the boundary layer and turbulent dispersion to be simulated. Some physico-chemical processes are also taken into account: chemical reactions  $\text{NO}-\text{NO}_2-\text{O}_3$ , transport and deposition of particles (dry deposition) and leaching of particles by rain (wet deposition).



**Figure 3:** Illustration of street-canyon volumes in SIRANE (source: ECL - LMFA).

SIRANE uses a large number of input data specific to the area under study: network of interconnected streets (see Figure 3), hourly evolution of the territory's emissions data (road, heating, agricultural, etc.), main meteorological variables (wind speed and direction, temperature, precipitation, solar radiation, etc.), background pollution level (external contribution to the area) and proximity pollution measurements from the territory's measuring stations.

From an execution and performance point of view, SIRANE is designed to adapt to the multiprocessor performance of today's computers and can therefore take advantage of HPC architectures, as in the AQMO project. It can manage a large number of input data and interactions (high building density, detailed road network, pollutant accumulation phenomena in confined streets) while providing high output resolution.

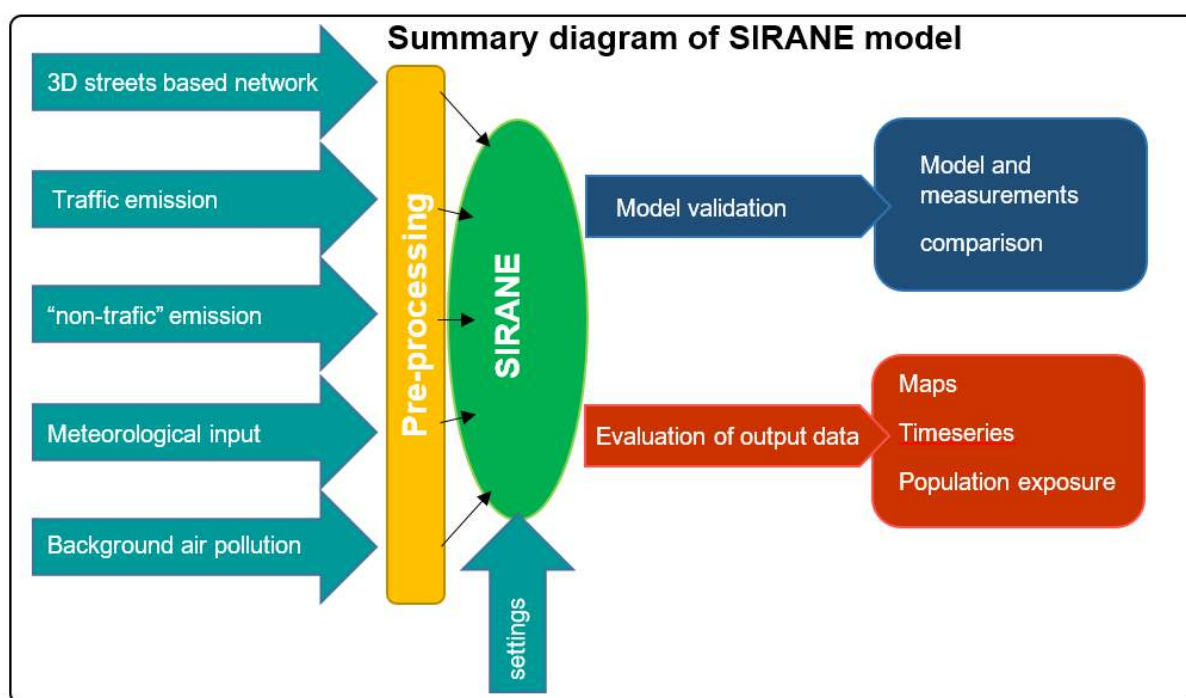
However, it is important to find a compromise between a high level of input data accuracy and reasonable computing performance. Moreover, SIRANE is limited, due to its methodology, in the processing of spatial scales that are too precise (street scale) or too wide (county scale).

The use of SIRANE is relevant within the framework of an urban model, from the scale of a city to the scale of a district.

Thus, SIRANE integrates theoretical models and simplified formulations capable of realistically representing the dispersion and the physico-chemical processes affecting the main atmospheric pollutants on the scale of one or more contiguous cities or a district.

### 3 Urban Model Methodology and Validation

In order for the urban models to be comparable with each other, it is necessary to set up a specific methodology to ensure the replicability of the models. The flowchart below (Figure 4) describes this methodology.



**Figure 4:** Process of creating an Air Breizh urban model;

The methodology implemented for the SIRANE model involves a certain amount of input data and parameterisation (depending on the area studied and the period under consideration). These data feed different software preprocessors in charge of preparing the data to feed the SIRANE model. The model allows obtaining sets of output data for the validation of the model (from the measurements made) and for the production of analysis data necessary for an air quality study.

#### 3.1.1 Creation of an optimised network of interconnected streets

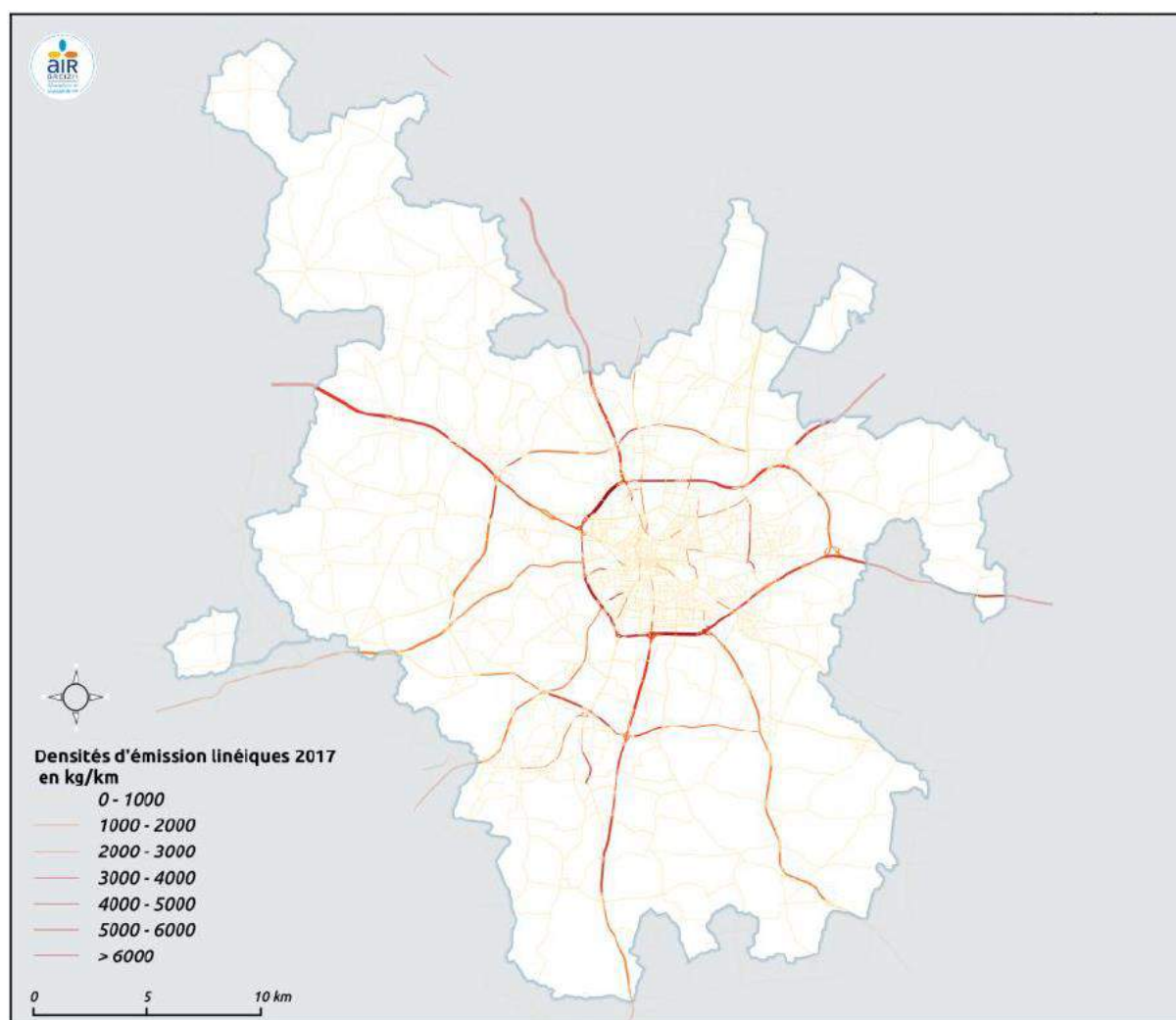
Depending on the area studied, the IGN databases or databases of the territory's road services (roads and buildings) are cross-referenced to identify the volumes of streets. This 3D street network will serve as a space for the "evolution" of the various pollutant dispersion phenomena (see Figure 2).



### 3.1.2 Calculation of pollutant emissions - related to road traffic

To simulate pollutant dispersion phenomena, it is necessary to know the emissions of these different pollutants in the area under consideration. This is the objective of the following stages of the methodology.

First of all, road traffic emissions must be identified. Indeed, in urban areas, these are among the most important emissions. They will be projected on the network of previously built streets. The calculation of these emissions is based on the COPERT V methodology [1]. To do this, it is necessary to know the annual volumes of vehicles circulating on the roads and their average speeds. Hourly changes in vehicle volumes on reference sections are also necessary in order to distribute the calculated annual emissions over time. The composition of the territory's vehicle fleet is taken from CITEPA statistics [2].



**Figure 5:** Emissions of nitrogen oxides  $\text{NO}_x$  from the road network in Rennes Métropole.

Local phenomena linked to certain sections cannot therefore be modelled using this methodology, as they would require too many IT resources. Moreover, these modulations do not take into account the variation in vehicle speed on the same section.



### 3.1.3 Calculation of pollutant emissions related to other sources

The calculation of "off-road" emissions is based on the spatialized cadastre of emissions built by the association. This tool makes an association between the nature of the soils (from CLC [3]) and different sources of information describing the following emissions: district heating, agricultural emissions, sea salts, etc.

In addition to these emissions described spatially by area, there are also emissions described on a case-by-case basis and associated with installations from the ICPE baseline, as shown in Figure 6. These emissions are timed over the year in order to take into account the reality of the area studied: periods of activity, meteorological conditions.

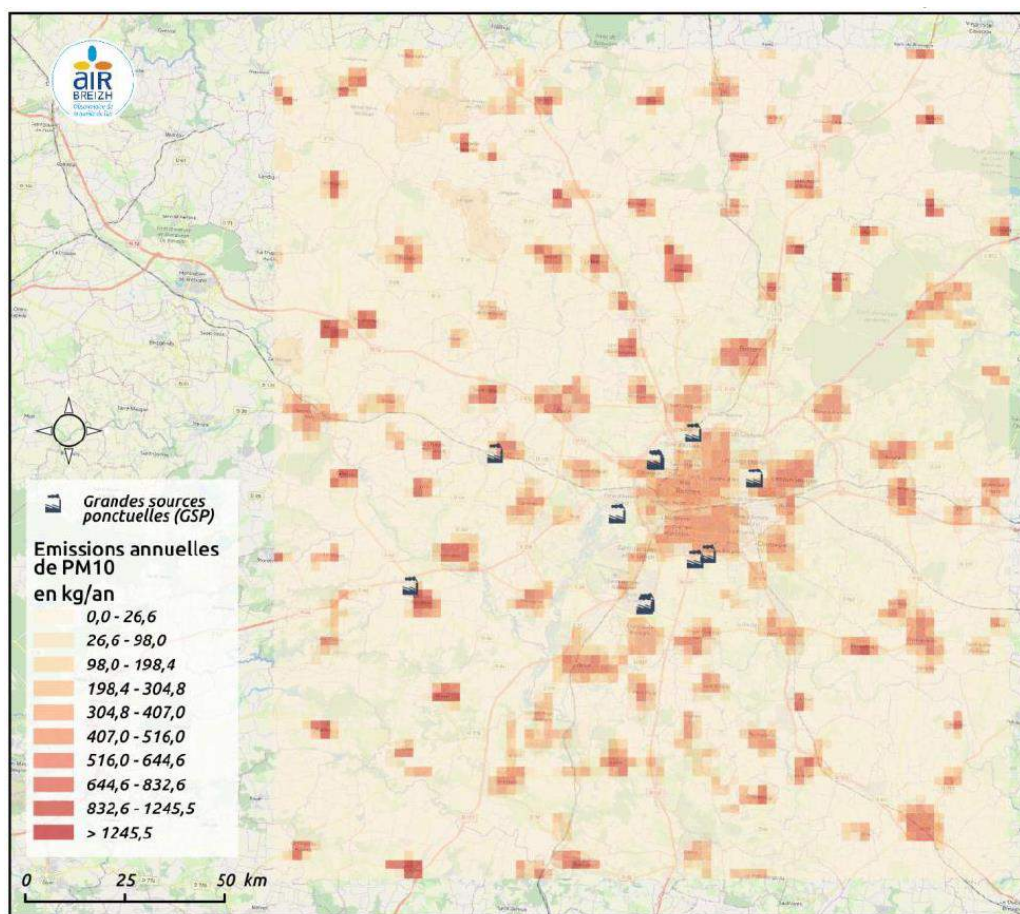


Figure 6: representation of 2016 annual PM10 emissions (residential sector).

### 3.1.4 Consideration of weather conditions

From a meteorological point of view, it is necessary to know a few basic parameters on the territory under consideration: the wind speed and direction that will allow the spatial and temporal evolution of the background pollution as well as the conditions of thermal stratification of the atmosphere (temperature, incident solar radiation, rainfall). These data come from stations or from the ARPEGE Météo France model.

### 3.1.5 Determination of background pollution

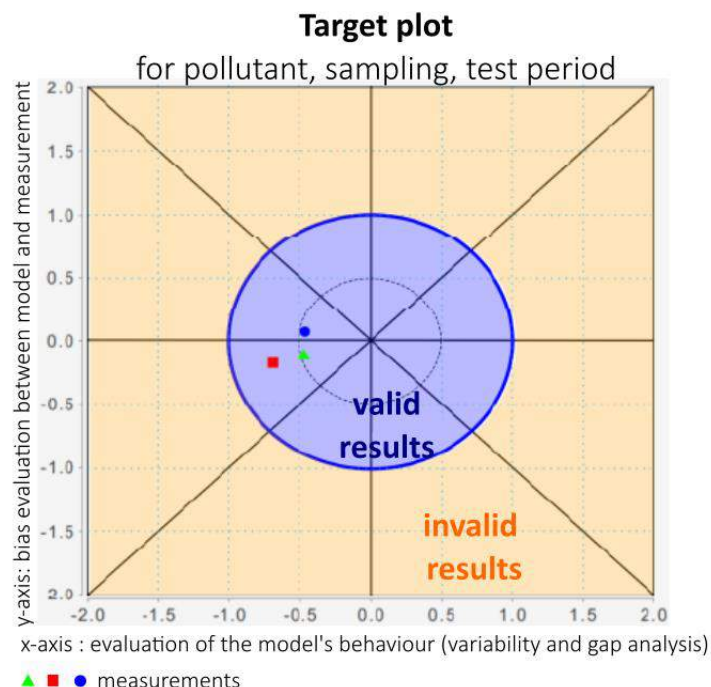
The background pollution represents the regional pollution that is measured outside of the studied area. It allows to associate to the model in progress, boundary conditions allowing to manage the influence of the outside on the study area. It can be influenced by multiple emission sources, including the urban and agricultural areas of the territory. In the SIRANE model, these pollutant concentrations are therefore added to those resulting from the emissions of the sources included in the domain (point, linear and surface).

## 3.2 Representation of the studied territory and validation of the model

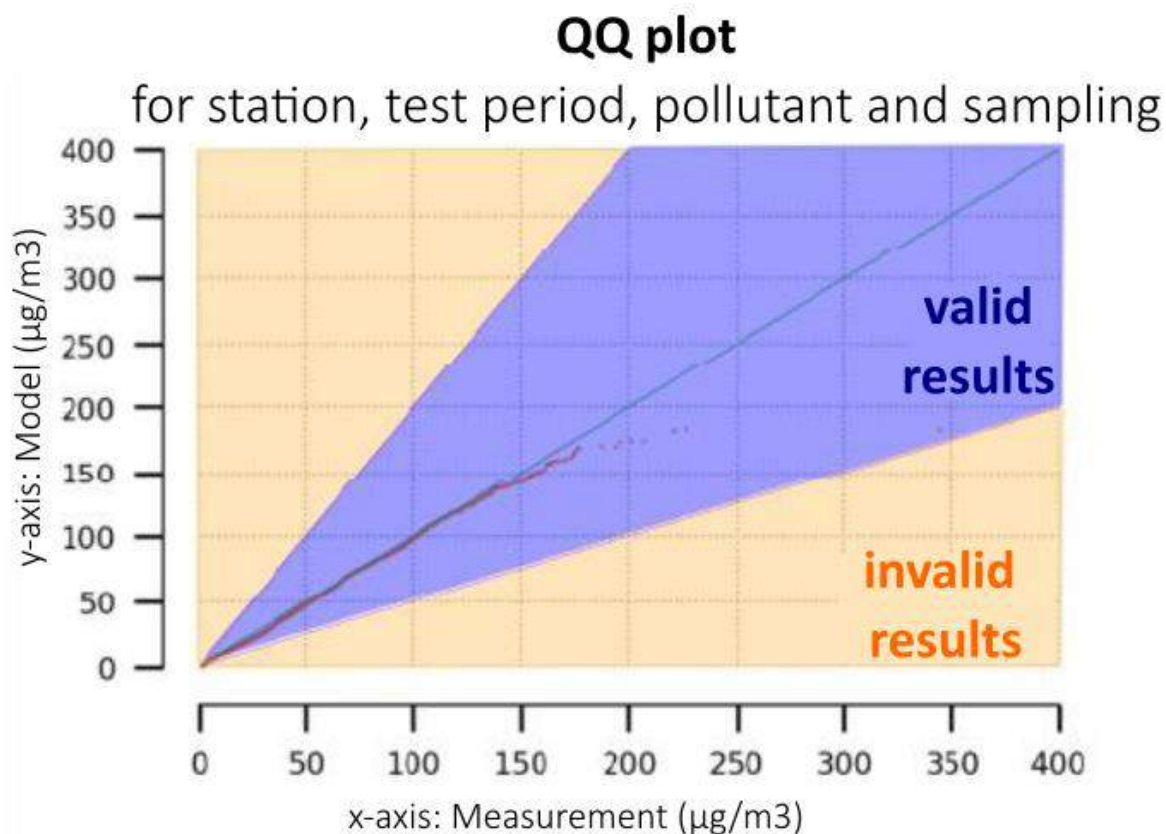
In order to have a "High Resolution" output, the SIRANE software describes the studied territory in the form of a grid of points distributed every 10m. The spatial resolution obtained (10m x 10m) makes the model particularly adapted to the scale of the district. On the other hand, one must therefore work with caution on finer scales (street scale for example). In addition, so-called "receiver" points are used to calculate the concentration of pollutants in order to analyse the performance and validity of the model at these predefined points.

The validity of the model is therefore calculated, at these receiving points, with a specific "Delta Tool" software package that allows the modelled concentrations to be compared with the measurements taken in the field, either at the stations of the association's network, or during specific studies carried out in the territory studied and during the period under consideration.

Each of these studies listed, and the measurements at the stations will be compared to the model.



**Figure 7:** Illustration and meaning of an evaluation target "Target Plot" from Delta Tool.



**Figure 8:** illustration and meaning of a Quantile-Quantile figure.

The "Delta Tool" [5] is used on a national scale by all air quality monitoring associations, thus allowing the same model validity criteria to be considered. In particular, the outputs of the software are produced in the form of two types of graphs: target (e.g. Figure 7) and reference line (e.g. Figure 8). These graphs make it possible to characterise the performance of the model in relation to in situ measurements.

### 3.3 Review of the methodology and its validation

The entire methodology described in the previous pages, as well as the SIRANE model on which it is based, are systematically detailed by the Air Breizh association in the various published modelling reports [7,8].

#### 3.3.1 Constraints on calculation performance vs. output precision

A "higher resolution" modelling implies that the model must be optimised to consider the processing and production time of the outputs. This is notably the case for 2 points: the necessary optimisation of the road network (improvement of calculation times while maintaining the necessary precision for population exposure calculations) and the calculation of population exposure (linked to the precision of the road network, of the distribution of the population in the buildings resulting from the MAJIC [6] data).

## 4 IDRIS – AQMO set up architecture and security constraints

This section presents first the IDRIS context, security regular usage and the next, compromise setup for AQMO architecture needs.

### 4.1 Context and regular usage

The Institute for Development and Resources in Intensive Scientific Computing (IDRIS), founded in 1993, is a centre of excellence in intensive numerical calculations, which serves the research branches of extreme computing. This concerns the application aspects (large-scale simulations) as much as the research inherent to high performance computation (calculation infrastructures, resolution methods and associated algorithms, processing of large data volumes, etc.).

IDRIS is the major centre of very high performance intensive numerical computation for the French National Centre for Scientific Research (CNRS). The principal objective of IDRIS is to contribute as effectively as possible to the excellence of scientific research in the modelling and intensive numerical calculation domain.

In order to accomplish these missions, IDRIS intervenes on two levels:

- 1) As a **service structure** through the installation and operation of a diversified avant-garde environment of intensive calculations, both versatile and evolving, and adapted to the great scientific challenges of numerical simulation. This environment encompasses a highly skilled user support team offering strong value-added services.
- 2) As a **transfer agent of the latest technologies** from computer science research and development towards the national infrastructures of high performance calculations. IDRIS not only provides assistance and training but is also involved in the development and optimisation of scientific codes. Situated at the intersection of science (numerical simulation) and technology (information systems), and in close communication with its scientific users, IDRIS is in a privileged position to progressively integrate new technologies into the national system of scientific research.

### 4.2 Usage modalities

The usage modalities of the IDRIS computing means are similar to those of the other large national equipment installations.

Requests for resources are made with the DARI form (*Demande d'Attribution de Ressources Informatiques*) through a [common web site](#) for the three computing centres (see [Requesting resource hours on IDRIS machines](#)).

The proposals are examined from a scientific perspective by the Thematic Committees who draw on the technical expertise of the centres' application assistance teams as needed. Subsequently, an evaluation committee meets to decide upon the resource requests and make approval recommendations to the Attribution Committee, under the authority of GENCI, for distributing computing hours to the three national centres.



Between two project calls, IDRIS management studies the requests for **supplementary resources as needed** (“demandes au fil de l’eau”<sup>1</sup>) and attributes limited hours in order to avoid the blockage of on-going projects.

### 4.3 Account management: account opening and closure

Each user has only one **account** that can be associated with all the projects in which the user participates.

All your account management is done by completing an [IDRIS form](#) which must be sent to [gestutil@idris.fr](mailto:gestutil@idris.fr).

An account can only exist as “open” or “closed”:

- **Open.** In this case, it is possible to:
  - Submit jobs on the compute machine if the project's current hours allocation has not been exhausted (cf. [idracct](#) command output).
  - Submit pre- and post-processing jobs.
- **Closed.** In this case, the user can no longer connect to the machine. An e-mail notification is sent to the project manager and to the user at the time of account closure.

#### 4.3.1 Opening a user account

##### For a new project

There is no automatic or implicit account opening. Therefore, each user of a project must request one of the following:

- If the user does not yet have an IDRIS account, the opening of an account respects the access procedures ([modalités d'accès](#)) defined by GENCI :
  - For **regular access** or **preparatory access**, use the [account creation request form](#) (GENCI) **after** the concerned project has obtained computing hours.
  - For **dynamic access**, you may create/access your private space throughout the year via the [DARI homepage](#), section “Développer des algorithmes en Intelligence Artificielle (IA<sup>2</sup>)” and the link “Création d’un compte ou connexion sur eDARI<sup>3</sup>” (in French).
- If the user already has an account open at IDRIS, he or she must complete the first section of the IDRIS [FGC](#) form, “Multiproject logins: Add a project”. This request must be signed by both the user and the manager of the added project.

**IMPORTANT INFORMATION:** By decision of the IDRIS director or the CNRS Defence and Security Officer (FSD), the creation of a new account may be submitted for ministerial authorisation in application of the French regulations for the [Protection du Potentiel Scientifique et Technique de la Nation](#) (PPST). In this event, a personal communication will

---

<sup>1</sup> On-demand requests

<sup>2</sup> Develop Artificial Intelligence Algorithms

<sup>3</sup> Create an account or log in on eDARI

be transmitted to the user so that implementation of the required procedure may be started, knowing that this authorisation procedure may take up to **two months**.

**Comment:** The opening of a new account on the machine will not be effective until (1) the access request (regular, dynamic, or preparatory) is validated (with ministerial authorisation if requested) and (2) the concerned project has obtained computing hours.

### For a project renewal

Existing accounts are automatically carried over from one project call to another if the eligibility conditions of the project members have not changed (cf. GENCI explanatory document for project calls [Modalités d'accès<sup>4</sup>](#)).

If your account was open and already associated with the project that obtained hours for the following call, no action on your part is necessary.

### 4.3.2 Closure of a user account

#### Account closure of an unrenewed project

When a GENCI project is not renewed, the following procedure is applied:

- On the date of project expiry:
  - DARI hours are no longer available and the project accounts can no longer submit jobs on the compute machine for this project.
  - However, the project accounts remain open to be able to access to their datas for a period of six months.

Account closure following expiry of the ministerial authorisation for access to IDRIS computer resources

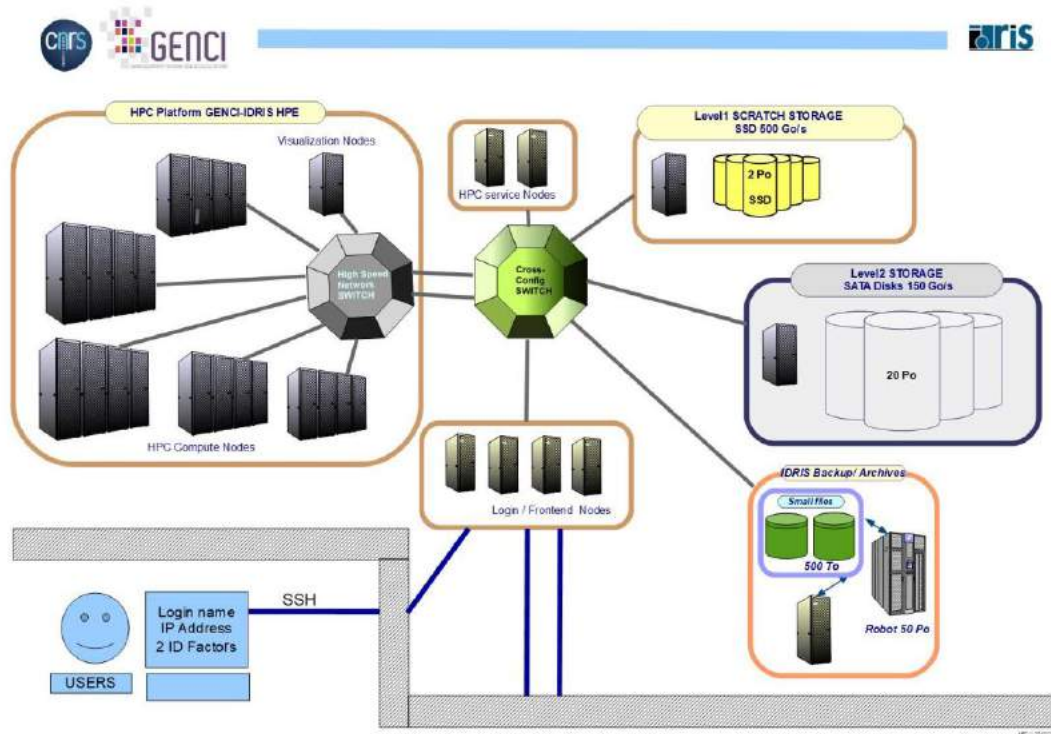
The ministerial authorisations are only valid for a certain period of time. When the ministerial authorisation reaches its expiry date, we are obligated to close your account.

---

<sup>4</sup> Access procedures

### 4.3.3 Access to the resources

IDRIS HPC infrastructure is composed of several elements, as depicted on Figure 9.



**Figure 9:** IDRIS HPC cluster architecture.

1. Frontend nodes: “entry point” of the cluster, users connect to these servers (e.g., through ssh) to submit jobs, manage their projects storage files, develop their simulation code, ...
2. Scheduler service nodes: act as a management node that hosts all the services in charge of the proper functioning of the cluster (e.g., the SLURM workload manager/job scheduler in charge of dispatching the jobs on the compute nodes).
3. Compute nodes: core of the cluster, on which simulations/data processing are executed. Compute nodes are heterogeneous in terms of hardware (e.g., with/without accelerators, types of CPU, amount of ram...)
4. Several levels of storages:
  - a. An small GPFS parallel filesystem, without any performance requirement, to hold the home directories of the users, and any “static data” (e.g., software installation directories)
  - b. A GPFS parallel file system (high performance) to hold short-lived working/scratch directories for I/O intensive operations
  - c. A GPFS parallel file system to hold long-lived working directories for non I/O intensive operations
  - d. A backup/archive system for long term storage
5. Visualization nodes and Pre-Post process nodes: designing models and viewing the results of the simulations can be data intensive and CPU/GPU intensive. Remote visualization nodes usually embed GPUs for efficient 2D/3D rendering and allow the users to visualize their data where they are generated instead of downloading GB/TB.

## 4.4 IDRIS - AQMO security architecture and usage

### 4.4.1 IDRIS WEB service portal architecture

IDRIS have had a previous portal technology experience with E-Biothon project, it was in a dedicated computing platform. AQMO needs to rethink the classic usage of a national academic HPC centre and the “IDRIS context and regular usage” had to be reconsidered, adapted to keep as close as possible the guideline of the centre security model. IDRIS is an AQMO resource provider. The “HPC WEB services Portal” links AQMO “Primary Server” and IDRIS HPC services (cf. Figure 10).

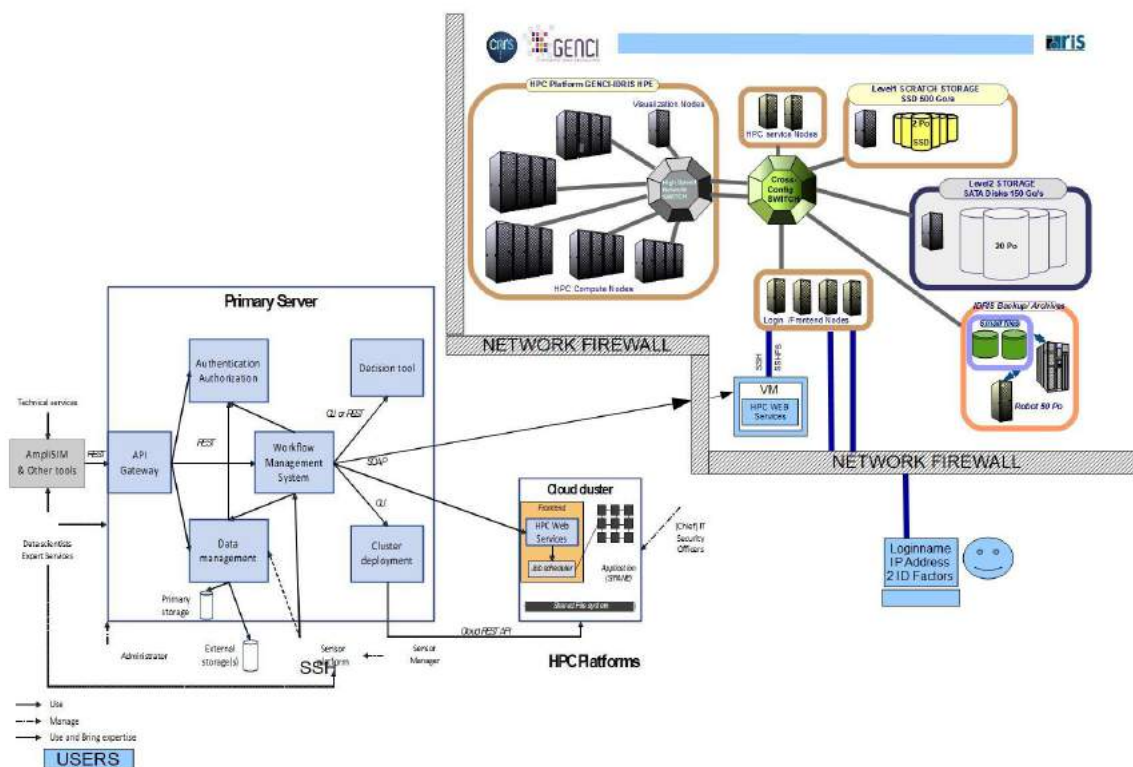


Figure 10: AQMO to IDRIS HPC architecture.

### 4.4.2 WEB service portal confinement

An IP and port filter is in place between the AQMO “Primary Server” and the “HPC WEB services Portal Server” and between the “HPC WEB services Portal Server” and the supercomputer Frontend nodes, in order to guarantee that only connections from the authorized host, on the authorized port, and using the authorized credentials, arrive to the front-end nodes.



#### 4.4.3 WEB service portal access

The whole process of receiving connections, performing simulations, storing temporary data and accounting for the used computing resources is conducted using a specific robot account ("*compte mandataire*"<sup>5</sup> also denominated *robot account*). All the users of the HPC Web Services tool use it collectively, but the security officer requires each AQMO user to have a pseudoanonymized personal account, where the identification is possible if needed.

A robot account is created within the context of an IDRIS project created specifically for AQMO, thus following the same procedures as any other IDRIS user and project. Every robot account has a physical person responsible for it, usually the IT Security Officer of the entity, which is using the account to interact with the HPC supercomputer front-end nodes. The robot account must have a strong password (12 chars, 3 char's categories) with a limited time of life (12 months) and a dictionary test during its setup process. A second authentication factor (OTP, U2F, X.509) is required for each user accessing the portal.

#### 4.4.4 WEB service portal account rights:

The IDRIS WEB service user account's rights are following some security rules:

- The less privilege setup
- The role setup (an admin account cannot use the Web service, and a Web service user cannot create or modify accounts)
- An up-to-date and documented account list must be maintained

#### 4.4.5 WEB service to HPC links specifications:

IDRIS WEB service is managed by EnginFrame, which is statically deployed on a VM in the HPC Centre (IDRIS). This VM has an indirect access to the SLURM job scheduler and the file systems of the supercomputer, across HPC frontend nodes. Wrappers are implemented on top of the job scheduler's commands to execute them through *passwordless* SSH and use SSHFS for folders and files listing.

SSH Force Command mechanism are in place for the SSH connections to IDRIS HPC frontend nodes, thus restricting effectively the catalogue of available commands for the robot accounts in the front-end nodes to those strictly necessary to perform the simulation.

SSHFS mount are performed with the read only and noexec flags, thus allowing the listing of user files and directories through the remote mount but not the execution of commands.

The frontend nodes receive SSH-based connections: SSH for command executions, SCP or SFTP for file transfers, and optionally SSHFS for folders and files listing. Then, the frontend nodes relay any simulation request to the compute nodes using the workload manager in place (e.g., SLURM in the case of Jean-Zay at IDRIS). The workload manager allocates the computing nodes if these are available, otherwise it will queue the request until so. The compute nodes execute the simulation software installed in the first level storage over the data transmitted previously to the frontend nodes (which is accessible to the compute nodes though the GPFS parallel file system). After the simulation ends, results are placed in the

---

<sup>5</sup> Proxy account

shared storage too, and they can be recovered back by the HPC Web Services tool through another SCP/SFTP connection to the front-end nodes.

#### **4.4.6 WEB service access tracking**

Any entity performing a direct connection to the supercomputer front-end nodes in order to submit jobs or access data through the HPC Web Services must log locally the actions performed, the actions triggered to the supercomputer (i.e. commands submitted) and the user responsible for those actions. Therefore, the entity must know the user who is profiting the HPC centre resources, even if such user is only interacting with a third-party service provider. These logs must be kept for at least a year.

### **4.5 IDRIS – AQMO setup process**

#### **4.5.1 WEB service portal setup process**

In order to evaluate if IDRIS could be a resource provider for AQMO, a preparatory phase has been organized involving IDRIS security officers and system staff on one side and UCit architects on the other side. Many meetings were necessary to converge towards some architecture scenarios and security overview.

Aside the core AQMO architecture, its portal foundations needed to be drawn and setup in order to allow UCit's architects to connect with privileges to the IDRIS web services server, in order to install and setup the EngineFrame framework. For internal security reasons, this setup is not precisely described in this document, but it consists in a series of IP filtered firewall gateways, requiring multifactor authentication, keeping detailed track of processes and commands executed; all unauthorized/suspicious access or actions raise alarms and are analysed by the IDRIS Security Operation Center.

#### **4.5.2 Simulation applications to IDRIS infrastructure adjustment**

IDRIS offered a user advanced support for the AmpliSIM's simulation codes. This service is usually opened to selected projects for limited time duration of 1-2 months renewable. For AmpliSIM's, the efforts covered the following area:

- codes porting and optimization, to take full advantage of Jean-Zay HPC specific architecture
- codes compilation support, libraries dependence identifications and installations.
- file user space organization accompaniment
- SLURM jobs user scripts and work chain setup support

#### **4.5.3 IDRIS – AQMO setup Rex**

If IDRIS have had a limited access portal experience within E-Biothon and DEISA projects, AQMO project forced the different technical actors (security, system and network) to be deeply involved to setup an architecture satisfying AQMO needs and IDRIS constraints. This collective effort represents in many points a strong model for the incoming IDRIS portal's architectures and setup.

#### **4.5.4 IDRIS – AQMO cost model**

Apart for some training opened to industrials, IDRIS usage is freely distributed through resources and storage allocation to scientific projects demands examined by Thematic

Committees and arbitrated under the authority of GENCI. It means that no charge is required to the users. This free access to resources does not avoid IDRIS CNRS guardianship and GENCI to be closely interested by the HPC centre cost model.

IDRIS center cost model is globalized and evaluated according the following items:

- HPC configuration cost including
  - Acquisition and installation cost
  - Storage acquisition and installation cost
  - HPC maintenance cost
  - Storage maintenance cost
- HPC center internal cost including:
  - Cooling system acquisition and installation cost
  - Cooling system maintenance cost
  - Power supply equipment's acquisition and installation cost
  - Power maintenance cost
  - Electricity Bill, which is an important cost factor and which is reduced using modern HPC configuration accepting hot water cooling system like for IDRIS jean-zay HPC configuration. In addition, the cooling system is more energy efficient, because it does not need compressors, but only adiabatic dry or free cooling.
- HPC center human resource cost, which can be sub-divided in 4 items
  - Administration staff, which includes the people in charge of payment, managing orders, administrative organization, ...
  - Level1 operation support staff, which provides 24h/7d detection and follow-up of material and software incidents
  - System – Network – Security staff, in charge of installing under a secured process all environment needing privileged access, giving support to other teams when needed
  - User Support staff, providing users assistance and training, giving support in the development and optimization of scientific codes, installing libraries and compilers

Once normalized all these items give a global cost which represents about 0,586€ for a GPU simulation hour processing and 0,62€ for a CPU simulation hour processing.

The IDRIS AQMO cost model can be divided in two parts:

1. The setup process preliminary step described in 4.5.1 with the “WEB service portal setup process” involving IDRIS System – Network – Security teams and the “Simulation applications to IDRIS infrastructure adjustment” involving IDRIS User Support team. For this step, the cost of academic specialized HPC engineers represents about 100,000 euros per year per person, with a cumulated engineer's participation representing about 2/3 person year.
2. AQMO simulation production step, which meets the global normalized HPC usage described before, and which depends on the amount of simulations executed and the architecture type used.

## 4.6 HPC as a service access at IDRIS

Access to HPC as a service is provided by web services exposed by the web portal (NICE EnginFrame) deployed on the **idraqmo** VM. Once an HPC service is defined in the portal (see Deliverable D4.1 - Section 6.3), it can be published by the administrator to selected groups of users. The AQMO platform can then access the services through SOAP web services.

Authentication to these web services is done through X509 certificates: user accounts are created in EnginFrame with a trusted certificate (EnginFrame manages the certificates through a Java truststore). When a user wants to access the service, he needs to connect with the certificate and the associated private key.

To simplify access to the SOAP APIs, a command line client has been developed. It abstracts the complexity of composing with the SOAP APIs by providing simple actions required to submit SIRANE simulations: files upload/download/list, services list/describe/execute, spoolers (object associated with a simulation containing jobs and input/output files) describe/list/download.

## 5 AWS set up architecture and security constraints

The Cloud infrastructure complements the computing resources provided by IDRIS. Depending on the use case (e.g., number and types of simulations...), user constraints (e.g., costs, time to result...) and computing centre's resources availability, Cloud resources can be used to run the simulations. These resources are transient and automatically deployed on demand in one or multiple virtual HPC clusters when a need arises.

### 5.1 Solution architecture overview

Cluster resources are created on demand in the AWS Cloud. The CCME solution (Cloud Cluster Made Easy) is deployed on a specific instance in AWS (the "Management Host"), and serves as the entry point to manage all clusters in the Cloud - see Figure 11.

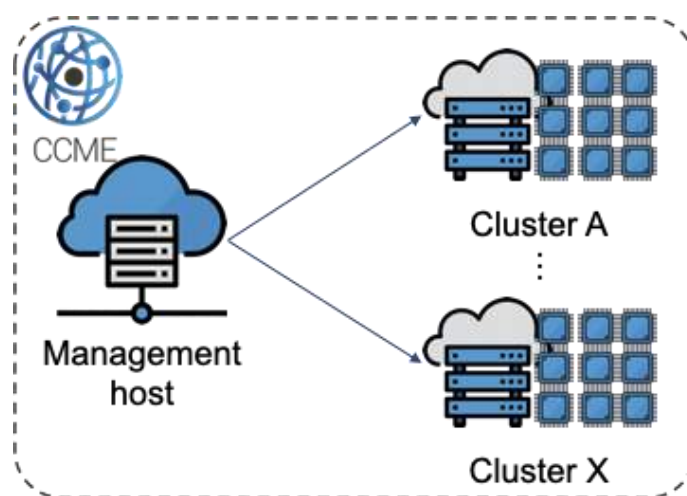


Figure 11: High-level view of the solution.

CCME provides a command line interface (CLI) to manage and interact with the clusters. It provides the capability to:

- create, update (only some elements of a running cluster can be updated, see next section) and delete a cluster
- list and get the status of running clusters
- start and stop the compute instances of a cluster to leave only the master node up and running
- define a time-to-live (TTL) for a cluster to automatically delete it when it is no longer needed
- ssh to a cluster

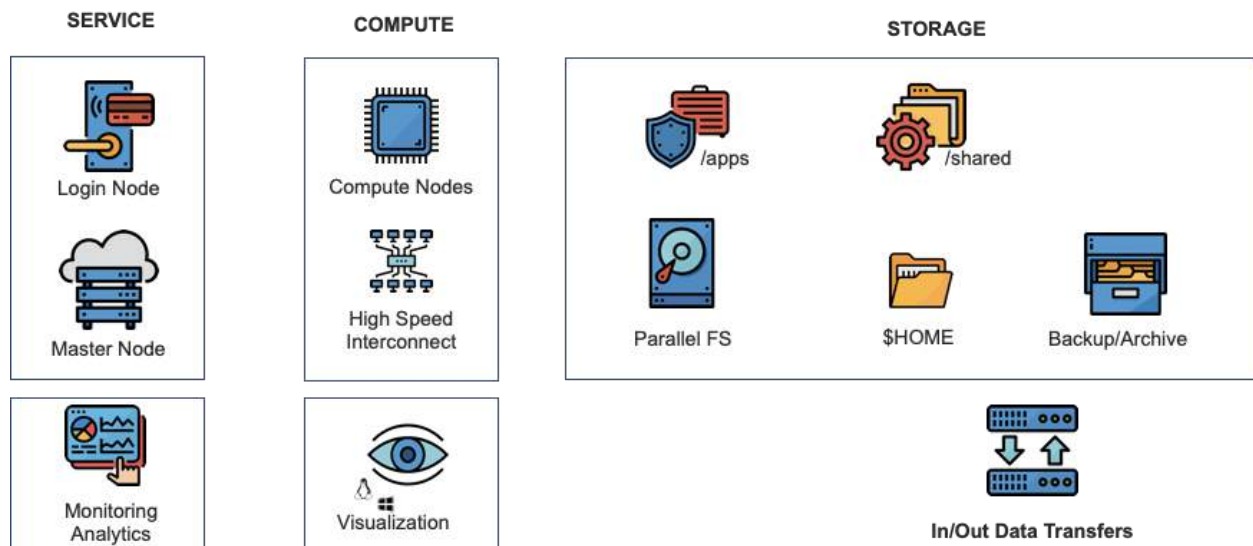
The complete lifecycle of a Cloud cluster can then be managed by the administrator of the platform, either manually or integrated in the Ryax workflow management system (number 9 in Figure 1).

Prior to any cluster deployment, CCME must be configured to define the available cluster profiles, i.e., the definition of the architecture of the clusters that will be deployed.

## 5.2 Cluster architecture

A on-premise cluster is composed of standard components that are depicted on Figure 12. Even if the manufacturers and performances vary from one cluster to another, these components can be found on any cluster:

- One or several services nodes that host the scheduler and associated services (e.g., cluster monitoring, databases...), and which act as the entry point for the end-users (usually directly through *ssh* or through a web portal)
- Some compute nodes interconnected by a high speed network for low latency communications between processes
- Optionally some visualization nodes with high-end GPUs for remote 2D/3D data visualization
- A hierarchy of storage systems ranging from a very high-throughput/low-latency parallel file system to backup and archival
- A strong network for cluster access and in/out data transfers



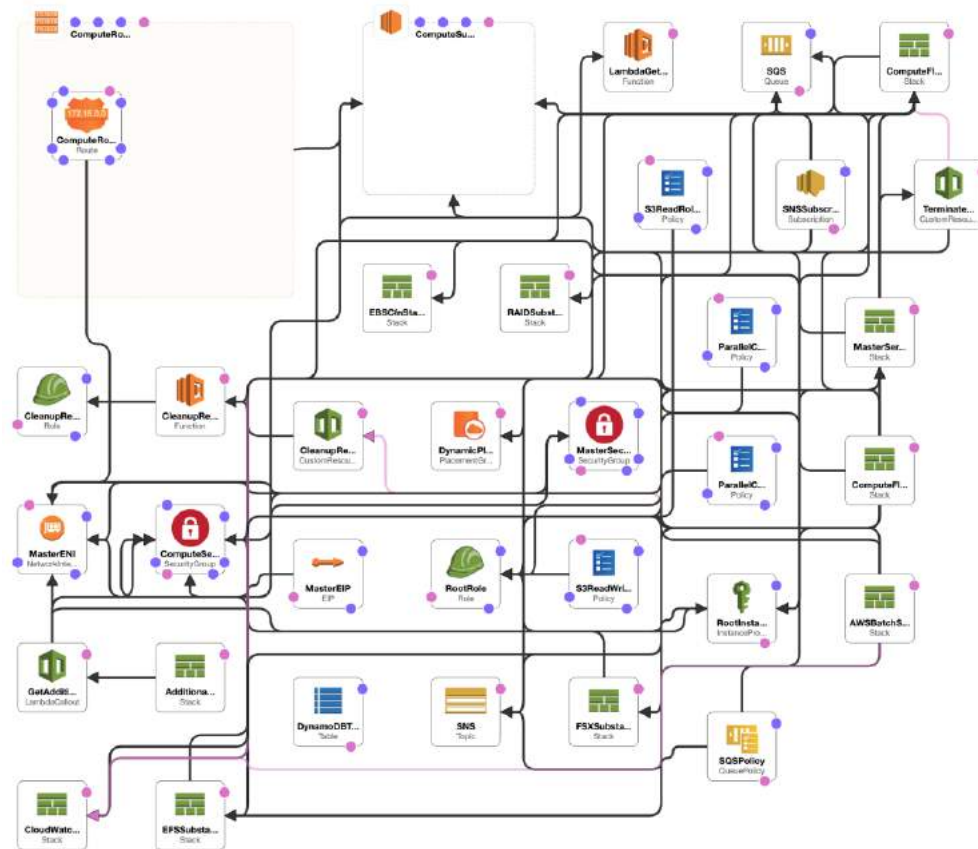
**Figure 12:** Cluster standard architecture.

A cluster in the Cloud follows exactly the same architecture, the difference being that all components are not directly accessible as hardware components, but as virtualized components accessible through web services. Here are a few examples of AWS services that are used to compose a cluster:

- Login/master node: EC2 (instances), EBS (disk storage), EIP (public IP addresses)...
- Compute nodes: EC2 (instances), EBS (disk storage), EFA (high speed network and interfaces), Autoscaling Groups (dynamic provisioning of instances)...
- Network: VPC (virtual private cloud - VLAN), SG (Security Groups - firewall)...
- Storage: FSx for Lustre (parallel file system), EBS (disk storage), S3 (object storage), Glacier (archive)...

Figure 13 presents the services (and their interactions) used in AWS to define and deploy an HPC Cluster.





**Figure 13:** AWS services used to build a Cloud cluster.

In CCME, the architecture of a cluster embeds all these components. The administrator can then select the exact definition and configuration of each of them, for example:

- The processor architecture (Intel, AMD, ARM)
- The type of instances used for the compute nodes (with more or less cores, RAM, local NVMe disks, network performances...), and their number (min and max number of compute nodes as the size of the cluster is dynamic and scales with the jobs' needs)
- The size and performance of the parallel file system

On top of these “hardware” characteristics, a cluster also embeds a set of software and system configurations:

- The type of job scheduler
- Configuration at the OS level (limits, packages, users...)
- The simulation software (SIRANE in our case)
- A web portal (NICE EnginFrame in our case) that presents resources and applications as web services

The whole definition of a cluster (hardware and software) is what we call a “cluster profile” in CCME. From a single profile, multiple clusters can be deployed, and multiple cluster profiles can be defined to suit different needs (e.g., a cluster with high-frequency compute nodes, but no high speed interconnect for data parallel applications; a cluster with a high speed

interconnect and a parallel file system for MPI applications...). When a cluster is created, the administrator selects a profile to associate with this cluster; later when the cluster is running, if needed he can update some of the parameters of the cluster to adapt to changing needs.

The deployment is piloted by an AWS service called CloudFormation that allows to define Infrastructure as Code: JSON templates define the services and their interactions, allowing automatic and repeatable deployments. The configuration on the master and compute nodes (e.g., package/software installation, OS configuration...) are done by Ansible scripts.

## 5.3 Security

Several layers are used to ensure the security of the solution, we detail them in this section: starting from the top (the AWS account), down to the network and web services.

First of all, the AWS account in which the solution is deployed is specific and dedicated to the AQMO project: no one outside this account can access the resources deployed in this account unless explicitly authorized. This account is part of an AWS Organization owned by UCit, and only has access to a subset of AWS' services through SCPs (Service Control Policies). Access to this account is allowed for a subset of administrators through IAM Roles (Identity and Access Management): these roles grant or deny access to a set of services and accounts through defined Policies. All administrators' access have a two factor authentication (TFA) activated, and a password policy that enforces password update every 90 days.

Within the AQMO account, the Management Host is deployed. Access to this instance is controlled by an AWS Managed Active Directory (AD) in which only a specified group of users can connect to the instance through ssh. From the management host, access to AWS services is further retrained through another IAM Role that limits any user to interact with only the AWS services needed to launch clusters (this Role prevents any privilege escalation). Network access to the Management Host is only allowed in SSH from a set of well identified IP addresses. This filtering is managed through Security Groups (SG) that allows communication within AWS to the clusters, and filters the protocols, ports and IP addresses from the Internet.

Access to the clusters follows the same principle: all clusters have an IAM Role attached to them restricting to the strict minimum the AWS services they can access, user management is done through groups in the AD, access to the master node is filtered through security groups that only allow ssh and https from a restricted set of IP addresses through SG, access to the compute nodes is only possible from the master node and between the nodes themselves.

All data in transit with the master node and the management host are encrypted (ssh, https), all communications between the compute nodes are not encrypted (for performance reasons for MPI processes). All data at rest is encrypted on the master node, the management host and in S3 (object storage), and can be encrypted on the compute nodes and the parallel file system (usually disabled for performance reasons) - encryption uses the Key Management Service (KMS) that securely generates, rotates and stores encryption keys.

The whole solution is deployed in a single Virtual Private Cloud (VPC), with two subnets: one for the Management Host and the master, and one for the compute nodes.



## 5.4 HPC as a service access

Just like the resources provided by IDRIS (see Section 4.6), the Cloud resources are accessible through Web Services and the wrapper developed on top of them to simplify the integration with Ryax workflow management system. See previous section about IDRIS for more details.

## 6 AmpliSIM service access API

The API provided by the AmpliSIM service allows to navigate among simulation results using web cartographic standards, including time dependant data. But the API offers also more in-depth manipulation capability to setup and launch simulations on-the-fly.

The access to the API is secured through https protocol and is using a user-based authentication token.

The service API is articulated around a simulation object. Attached to a simulation object are users, status, calculation clusters etc. but also simulation input data and simulation output data, including meta data. The API is hence available through three main components to manipulate:

- The simulation: creation, duplication, execution, etc.
- The input data attached to a simulation: creation, selection, modification, etc.
- The Output data attached to a simulation: consultation

Full details on the API documentation is provided in the Annex 1.

## 7 Data sensors integration and visualization

### 7.1 Integration

Sensor data acquisition is performed through automatic API access, in the same way meteorological or background concentration are accessed.

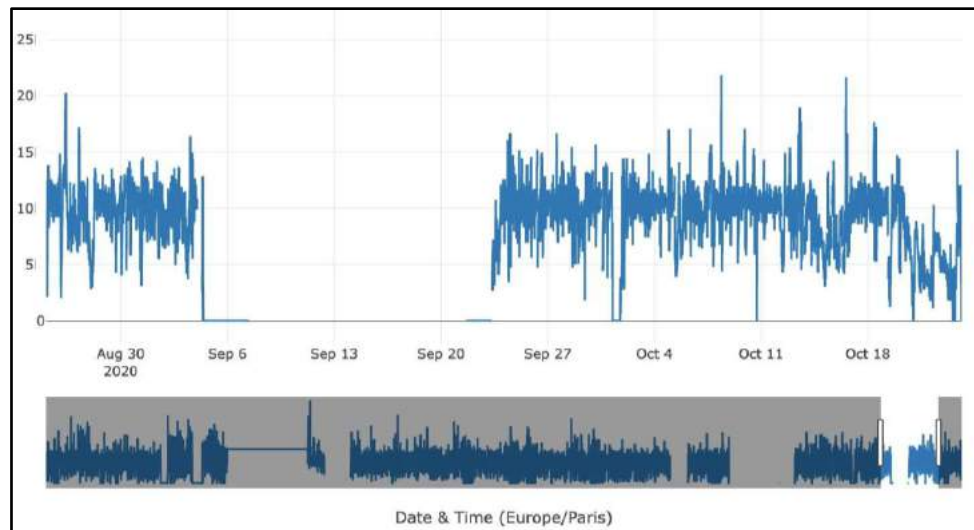
The AmpliSIM Web Modeling Service (AWMS) can access data based on API and token authentication. A library of API providing data is defined in the AWMS, and associated for each with a configuration file detailing the precise set of data to be accessed (station ID, locations, etc.). The Air Breizh API and the AQMO API have been included in the library of data to be retrieved. The configuration file for the relevant stations corresponding to the Rennes Metropolis Area have been set-up.

External data required for simulations are automatically downloaded by the AWMS to be available locally to the modelling users. This is performed with a scheduler launching asynchronous tasks, per day, week or month depending on the refresh pace of the source. Such data could be accessed purely on-demand. Nonetheless, and with the exception of extensive data such as surface raster data, experience has shown that multiple API access can be resource intensive and less stable. Multiple API access than can be common to multiple user benefit drastically from local cache availability. And this cache offers a security in case of downtime of the API provider.

## 7.2 Visualization

All sensor data available within the web platform can be displayed (e.g. Figure 14). The display is provided through the use of the Plotly Javascript open-source graphing library (see [9]).

Like meteorological data, or source emission data, concentration measurement can be displayed as time dependant data.



**Figure 14:** Plot of the time evolution of the NOx measurements at a sensor location.

The platform includes also manipulation for *datetime* data. Default display is UTC, but the user can chose a specific *timezone*. All displays are then using the *timezone* defined by the user.

## 8 Conclusion

The work described in this report allows providing simplified access to pollutant dispersion numerical simulation. The proposed solution not only allows computing resources access and management but also proposes an API to access that hides all complexity in setting up such numerical simulation. As a result, a generalization of the use of numerical simulation is possible in the context of digital twins for smart-cities. A proposal to extend the availability of this work at the national level is being discussed with GENCI (Grand Équipement National de Calcul Intensif).

## ANNEXE 1: AmpliSIM simulation service API

Filter... x

**Simulation**

Duplicate a simulation  
get simulation status  
launch simulation

### AmpliSIM-API

AmpliSIM simulation API

1.0.0

#### Simulation

##### Simulation - Duplicate a simulation

1.0.0

**POST**

`https://www.amplisim.com/simu/API/duplicate/{idsimu}/`

Permission : csrfToken

Python example usage:

```
from requests import session
urlCSRF = "https://www.amplisim.com/airquality/user/login/"
client = session()
client.get(urlCSRF)
csrfToken = client.cookies['csrfToken']
cookies = { "csrfToken" : csrfToken }
headers = {
    "X-CSRFToken" : csrfToken,
    "Referer" : urlCSRF,
    "ASSERVICETOKEN": "token provided by AmpliSIM",
}
urlDuplicateSimu = "https://wwwinput.amplisim.com/airquality/simu/API/duplicate/101/"
response = client.post(
    urlDuplicateSimu,
    timeout = ( 10, 10),
    headers = headers,
    cookies = cookies
)
```

**Cookie**

Champ	Type	Description
csrfToken	string	client csrf token from client session

**Header**

Champ	Type	Description
ASSERVICETOKEN	string	token provided by AmpliSIM
X-CSRFToken	string	client csrf token from client session (same as in Cookie)
Referer	string	url csrf login adress

**Paramètre**

Champ	Type	Description
idsimu	Number	Simulation unique ID to be duplicated.

**Success 200**

Champ	Type	Description
idsimu	Number	created simulation id

Success-Response:

```
{
  "idsimu": 4712
}
```

**Error 4xx**

Nom	Description
403	User not allowed to duplicate simulation
404	Unknown simulation id to duplicate

## Simulation - get simulation status

1.0.0

Simulation may have the following status values:

Short description	Description
Created	Simulation is created
Ready	Simulation is ready to be launched
Submitted	Simulation has been submitted to calculation server
Started	Simulation has started on calculation server
Completed	Simulation is completed
SimulationError	Simulation ended in error
LaunchError	simulation launch error
InputUploaded	Input files have been uploaded on calculation server
OutputDownloaded	Simulation output files have been downloaded
InputUploadError	Error upload simulation input files on calculation server
OutputDownloadError	Error downloading simulation output files from calculation server

GET

```
https://www.amplisim.com/simu/API/status/{idsimu}
```

Permission : token

## Header

Champ	Type	Description
ASSERVICETOKEN	string	token provided by AmpliSIM

## Paramètre

Champ	Type	Description
idsimu	Number	Simulation unique ID to be updated.

## Success 200

Champ	Type	Description
simulationStatus	Number	Simulation status short description
statusDescription	string	

Success-Response:

```
{
  "status":00,
  "message":"created"
}
```

## Error 4xx

Nom	Description
401	User not authenticated
403	User not allowed to retrieve simulation status
400	Error retrieving simulation status

## Simulation - launch simulation

1.0.0

POST

https://www.amplisim.com/simu/API/launch/{idsimu}

Permission: csrftoken

Python example usage:

```

from requests import session
urlCSRF = "https://www.amplisim.com/airquality/user/login/"
client = session()
client.get(urlCSRF)
csrftoken = client.cookies['csrftoken']
cookies = { "csrftoken" : csrftoken }
headers = {
    "X-CSRFToken" : csrftoken,
    "Referer" : urlCSRF,
    "ASSERVICETOKEN": "token provided by AmpliSIM",
}
data = {
    "datedeb" : "2018-05-30T00:00:00z",
    "datefin" : "2018-06-19T23:59:59z"
}
urllaunchsimu = "https://www.amplisim.com/airquality/simu/API/launch/101/"
response = client.post(
    urllaunchsimu,
    timeout = ( 10, 10),
    json = data,
    headers = headers,
    cookies = cookies
)

```

## Cookie

Champ	Type	Description
crsftoken	string	client csrf token from client session

## Header

Champ	Type	Description
ASSERVICETOKEN	string	token provided by AmpliSIM
X-CSRFToken	string	client csrf token from client session (same as in Cookie)
Referer	string	url csrf login adress

## Paramètre

Champ	Type	Description
idsimu	Number	Simulation unique ID to be updated.

## Success 200

Champ	Type	Description
status	string	message defining request status Valeurs autorisées: "error", "success"
message	string	Error or Success message

Success-Response:

```

{
  "status": "success",
  "message": "simulation launched"
}

```

## Error 4xx

Nom	Description
400	Error launching simulation
403	User not connected or not allowed to update simulation status

Généré avec apidoc 0.20.1 - 2020-10-30T17:28:35.480Z

Filter... x

SimulationInputs

Update simulation calculation period

# AmpliSIM-INPUT-API

AmpliSIM input data simulation API

## SimulationInputs

### SimulationInputs - Update simulation calculation period

1.0.0

POST

https://wwwinput.amplisim.com/api/v1.0/chrono/updateSimulationDates/{idsimu}/

Permission : csrf token

Python example usage:

```

from requests import session
urlCSRF = "https://www.amplisim.com/airquality/user/login/"
client = session()
client.get(urlCSRF)
csrf_token = client.cookies['csrf_token']
cookies = { "csrf_token" : csrf_token }
headers = {
    "X-CSRFToken" : csrf_token,
    "Referer" : urlCSRF,
    "ASSERVICETOKEN": "token provided by AmpliSIM",
}
data = {
    "datedeb" : "2018-05-30T00:00:00z",
    "datefin" : "2018-06-19T23:59:59z"
}
urlupdateChrono = "https://wwwinput.amplisim.com/airquality/updateSimulationDates/101/"
response = client.post(
    urlupdateChrono,
    timeout = ( 10, 10),
    json = data,
    headers = headers,
    cookies = cookies
)

```

#### Cookie

Champ	Type	Description
crsftoken	string	client csrf token from client session

#### Header

Champ	Type	Description
ASSERVICETOKEN	string	token provided by AmpliSIM
X-CSRFToken	string	client csrf token from client session (same as in Cookie)
Referer	string	url csrf login adress

#### Paramètre

Champ	Type	Description
idsimu	Number	Simulation unique ID to be duplicated.

#### RequestBodyParameters

Champ	Type	Description
datedeb	string	UTC beginning date of simulation, format "YYYY-mm-ddTHH:MM:ssz"
datefin	string	UTC end date of simulation, format "YYYY-mm-ddTHH:MM:ssz"

RequestBodyParameters-Example:

```

{
    "datedeb": "2019-01-02T03:00:00z",
    "datefin": "2019-01-02T03:00:00z",
}

```

Success-Response:

```

{
    "status": "success",
    "message": "OK"
}

```

#### Error 4xx

Nom	Description
403	User not allowed to update simulation chrono data
400	Unable to decode request body data

Généré avec apidoc 0.20.1 - 2020-10-30T17:05:01.627Z

Filter...

**SimulationOutputs**

- Get simulation concentration time series at a given key point
- Get simulation output metadata description
- Get tile at a given zoom level for a given date, field and vertical level

## AmpliSIM-OUTPUT-API

AmpliSIM simulation output API

1.0.0 -

### SimulationOutputs

SimulationOutputs - Get simulation concentration time series at a given key point 1.0.0 -

**GET**

```
https://wwwdata.amplisim.com/simdata/api/v1.0/simdata/keyPtsTS/{tdsimu}/{ptsid}/{encodingOutput=false}
```

Permission : token

Python example usage:

```
from requests import session
client = session()
headers = {
    "ASSEVICETOKEN": "token provided by AmpliSIM",
}
cookies = None
# Ask for first tile
urlGetKeyPts = "https://wwwdata.amplisim.com/airquality/api/v1.0/simdata/keyPtsTS/201/1/?encodingOutput=false"
response = client.post(
    urlGetKeyPts,
    timeout = ( 10, 10 ),
    headers = headers,
    cookies = cookies
)
```

**Header**

Champ	Type	Description
ASSEVICETOKEN	string	token provided by AmpliSIM

**Paramètre**

Champ	Type	Description
tdsimu	Number	Simulation unique ID to retrieve outputs description.
ptsid	Number	Key point id to extract
encodingOutputFlag	String	to encode as 64 bits time serie output or not Valeurs autorisées: "false", "true"

**Success-Response:**

```
{
  "status": "success",
  "message": "OK",
  "cs": {
    "data": {
      "unit": {
        "NOx": "ug/m3",
        "SO2": "ug/m3"
      },
      "ta": {
        "datetime": ["2019-01-01 12:00:00", "2019-01-01 13:00:00", "2019-01-01 14:00:00"],
        "NOx": [0.0, 1.0, 20.0],
        "SO2": [1.0, 5.0, 50.0]
      }
    }
  }
}
```

If encoded output is "true", "ta" object is encoded in 64bits to fasten response transfer.  
Datetimes are expressed with TimeShiftUTC nb hours from UTC (see simulation meta data description)

**Error 4xx**

Nom	Description
403	User not allowed to retrieve simulation output.
400	bad request



## SimulationOutputs - Get simulation output metadata description

1.9.0

GET

https://www.data.amplisim.com/simdata/api/v1.0/simdata/visu/{idsimu}/

Permission: token

Python example usage:

```
from requests import session
client = session()
headers = {
    "ASSEVICETOKEN": "token provided by AmpliSIM",
}
cookies = None
# Ask for first tile
urlGetMetaDatas = "https://www.data.amplisim.com/airquality/api/v1.0/simdata/visu/193/"
response = client.post(
    urlGetMetaDatas,
    timeout = ( 10, 10 ),
    headers = headers,
    cookies = cookies
)
```

## Header

Champ	Type	Description
ASSEVICETOKEN	string	token provided by AmpliSIM

## Paramètre

Champ	Type	Description
idsimu	Number	Simulation unique ID to retrieve outputs description.

## Success 200

Champ	Type	Description
version	String	output version number
simu	int	simulation id
keypts	object[]	list of key point description
id	Number	key point id
name	String	key point name
x	Number	key point x coordinate
y	Number	key point y coordinate
h	Number	key point height above ground
epsg	Number	key point geoprojection EPSG code
hasListing	Boolean	flag of presence of an excel listing file
minmax	object	min and max values per output fields name
keyPtsOutput	object	output values per key point id
header	object	list of header field in key point time-series files
CRU	object	list of pollutants available at key points for time series
site	String	geoson simulation site border
TimeShiftFromUTC	Number	nb of hour shift from UTC time in model outputs (>0 East of greenwich, <0 West)
webtiles	Object	description of output field available as tiles images
dates	Number[]	list of available dates (if datetime values, expressed with TimeShiftFromUTC from UTC)
min_zoom	Number	min zoom level available for tiles
max_zoom	Number	max zoom level available for tiles
fields	String[]	list of available output field
fieldinfo	Object[]	list of field description (same order than fieldinfo)
levels	Number[]	list of output vertical level ids available
scales	Object[]	color scale for each output field
name	String	output field name
legend	String	associated legend
unit	String	associated output unit
dataencoding	Number-1	tile encoding : 1 for images
scale	Object[]	list of scale color/values/label. The value for a given color is the lowest output value



Success-Response:

```
{
  'version': '1.1',
  'time': 288,
  'protocol': 0,
  'hostname': 'https://tutodata.amplisim.com',
  'keypts': [
    {
      'id': 1,
      'name': 'Rochefort Centre',
      'x': 893807.8,
      'y': 522584.8,
      'h': 8.8,
      'wpag': 32632
    }
  ],
  'hasListing': True,
  'minmax': {
    'SO2_CONCEN_03000': {'Max value': 0.01388824024530422, 'Min value': 0.051084117833001e-18},
    'SO2_CONCEN_C_03000_000_00': {'Max value': 0.4689418145175749, 'Min value': 1.878744694338163e-05},
    'SO2_DEPTOT_03000': {'Max value': 0.00643849391796284, 'Min value': 3.896649824481274e-12},
    'SO2_DEPTOT_C_03000_000_00': {'Max value': 0.002816038755039870, 'Min value': 8.472443241090007e-12}
  },
  'keyPtsOutput': {
    'i': [
      ['SO2', '2020', 1, 'Rochefort Centre', 0.892612073883126378, 'mg/m3', 'Hourly Mean Concentration']
    ],
    'header': [
      'Pollutant',
      'Period',
      'rcp_index',
      'rcp_name',
      'Value',
      'Unit',
      'ScatField'
    ],
    'CRD': [
      'SO2'
    ]
  },
  'site': {
    'type': 'Polygon',
    'coordinates': [
      [
        [
          5.581975335982629, 47.99285138499871
        ],
        [
          5.583183154287295, 47
        ]
      ]
    ],
    'TimeShiftFromUTC': 0,
    'nbTimes': {
      'dates': ['2020'],
      'fields': [
        'SO2_CONCEN_03000'
      ],
      'fieldInfo': [
        {
          'pollutant': 'SO2',
          'legend': 'Hourly SO2 Conc.',
          'legendshort': 'Hourly Conc.',
          'duree': '03000',
          'type': 'CONCEN'
        }
      ]
    },
    'levels': ['1'],
    'scales': [
      {
        'name': 'SO2_CONCEN_03000',
        'legend': 'Hourly SO2 Conc.',
        'scale': [
          {
            'label': '5.64e-03 ug/m3',
            'value': 0.005643643222213857,
            'color': '#978F12'
          },
          {
            'label': '0.01 ug/m3',
            'value': 0.01117048637811523,
            'color': '#90CF40'
          },
          {
            'label': '0.02 ug/m3',
            'value': 0.0167457395240186,
            'color': '#ACAD38'
          },
          {
            'label': '0.03 ug/m3',
            'value': 0.022327392609921974,
            'color': '#84EFC4'
          },
          {
            'label': '0.03 ug/m3',
            'value': 0.02777045815825346,
            'color': '#FEFF1F'
          },
          {
            'label': '0.03 ug/m3',
            'value': 0.03338469896172672,
            'color': '#F2071C'
          },
          {
            'label': '0.04 ug/m3',
            'value': 0.03883831118783259,
            'color': '#F3AF19'
          },
          {
            'label': '0.04 ug/m3',
            'value': 0.0443730852533546,
            'color': '#E88710'
          },
          {
            'label': '0.05 ug/m3',
            'value': 0.049945658398436835,
            'color': '#E95F13'
          }
        ],
        'dataencoding': 1,
        'unit': 'ug/m3'
      }
    ],
    'copyright': 'Copyright Amplisim 2020'
  }
}
```

Error 4xx

Nom	Description
483	User not allowed to save simulation chrono data
489	Unable to decode request body data

Error-Response:

```
{
  'status': 'error',
  'message': 'Error, request does not contain necessary informations'
}
```

## SimulationOutputs - Get tile at a given zoom level for a given date, field and vertical level

1.0.0

GET

<https://wwwdata.aplisis.com/simdata/api/v1.0/simdata/viewdata/TS/{idaimu}/{field}/{date}/{k}/{x}/{xtile}/{ytile}>

Permission: token

Python example usage:

```
from requests import session
client = session()
headers = {
    "ASSERVICETOKEN": "token provided by AmpliSIM",
}
cookies = None
# Ask for first tile
urlGetTile = "https://wwwdata.aplisis.com/simdata/api/v1.0/simdata/viewdata/TS/101/S02_CONCEN_03600/2020/1/12/211"
response = client.post(
    urlGetTile,
    timeout = ( 10, 10),
    headers = headers,
    cookies = cookies
)
#-----
# get asts from initial tile response in cookies
#-----
asts = response.cookies['asts']

#-----
# add it to all next tile requests
#-----
headers['ASTSTOKEN'] = asts
#
urlGetTile = "https://wwwdata.aplisis.com/simdata/api/v1.0/simdata/viewdata/TS/101/S02_CONCEN_03600/2020/1/12/211"
response = client.get(
    urlGetNextTile,
    timeout = ( 10, 10),
    headers = headers,
    cookies = cookies
)
```

### Header

Champ	Type	Description
ASTSTOKEN	string	token provided in the cookies of the response to the first request of tile image. To be added in following tile requests header to manage session connexion
ASSERVICETOKEN	string	token provided by AmpliSIM
X-CSRFToken	string	client csrf token from client session (same as in Cookie)
Referer	string	url curl login adress

### Cookie

Champ	Type	Description
csrftoken	string	client csrf token from client session

### Paramètre

Champ	Type	Description
idaimu	Number	Simulation unique ID to retrieve outputs description.
field	String	Output variable to plot (should be in available fields, see metadata output)
date	Number	year or datetime to plot (same format as in metadata output)
k	Number	vertical level id to plot (should be in available vertical levels, see metadata output)
z	Number	zoom level to extract (should be in available zoom levels, see metadata output)
xtile	Number	x coordinate of the tile to extract, tms system reference
ytile	Number	y coordinate of the tile to extract, tms system reference

### Success 200

Champ	Type	Description
image	Object	in PNG format

### Error 4xx

Nom	Description
403	User not allowed to save simulation chrono data
500	No data for the given tile coordinates