European Commission

HORIZON 2020

The EU Framework Programme for Research and Innovation

CURE System Requirements

Deliverable D4.1

DATE
31 March 2021

ISSUE
1.0

GRANT AGREEMENT
no 870337

DISSEMINATION LEVEL
PU

PROJECT WEB-SITE
http://cure-copernicus.eu/

LEAD AUTHOR
Mario Dohr (GeoVille)

CO-AUTHORS
Samuel Carraro (GeoVille)
Johannes Schmid (GeoVille)

CONTENTS

LIST OF TABLES

LIST OF ACRONYMS

App	Application
API	Application Programming Interface
CPU	Central Processing Unit
CCSI	Copernicus Core Service Interface
DIAS	Data and Information Access Services
RAM	Random-Access memory
Regex	Regular expression
WP	Work Package

1 INTRODUCTION

1.1 Purpose of the document

Deliverable 4.1 is the final delivery of Task 4.1 User requirements for application transformation and deployment of work package (WP) 4 Cure System Development. The purpose of the CURE System Requirements Specification is to transform the user requirements as well as requirements derived from App developers into system requirements for the CURE System.

This document is also serving as an input for the Deliverable 4.2 CURE System Design, which will be delivered at M18.

2 System Requirements Workflow

2.1 Overall approach and methodology

This document lay the baseline for the CURE System Design and the follow-up development.

To achieve this, dedicated user requirement meetings and workshops have been held in WP1 for discussion and consolidation. Detailed information was compiled in D1.1 Summary of User Requirements.

These user requirements served as an input for the App Developers in WP3 for the development of their individual applications and more details on the stakeholder needs addressed by each application are provided in Deliverable D3.1. A questionnaire was originally circulated to the App developers within WP6 (documented as an Appendix in Deliverable D6.1) and updated for WP4 (documented as an Appendix in this document) was used to identify the Requirements from the App developers on the System.

2.2 CURE User requirements questionnaire

The Apps Questionnaire, among other things, had been designed to collect baseline information to consider all hardware and software, as well as data requirements of the applications and enable transformation and deployment on a DIAS.

The relevant Questions to the System Design are:

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.

- What data does your service need?
- Does your code run with raw input data or does it require preprocessed data?
- How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)
- Does your service contain more than one task (series of script executions)?
- What resources does your service need for processing one unit (Cores, RAM)?
- Do you use Docker?

A copy of the full questionnaire responses, as updated during the time of this deliverable, is provided in the Appendix.

3 SYSTEM REQUIREMENTS

3.1 Summary from technical Part of the Questionnaire

Table 1 Summary of Questionnaire

Service	Input data	Input data state	Storage interim	Storage output	№ tasks	CPU'S	RAM (GB)
APP01 Local scale surface temperature dynamics	• Thermal infrared (TIR) measurements (1 km) • Atmospheric information (water vapor) • Surface Cover in 2 m spatial resolution • Red and InfraRed (NIR) measurements (10 m Sentinel-2)	Preprocessed	0	100s MB's	> 1	1	>=16
APP02 Surface UHI assesment	• CLMS HR Imperviousness 2018 • Land Surface Temperature from App01 • perhaps L2A (<60%cc) or yearly NDVI-Max	Raw and Preprocessed	< 10 GB	MB's	> 1		4
APP03 Urban Heat Emissions Monitoring	• LST from App01 • high resolution LULC map • high resolution DOMs (trees and buildings) • standard meteorology • Flux tower data	Preprocessed		36 kB	1		
APP04 Urban CO2 emissions monitoring	• High resolution LULC map • high resolution DOMs (trees and buildings) • standard meteorology • Diurnal/seasonal courses) of source strengths • Flux tower data • Footprint climatology	Preprocessed		1 GB			

APP05 Urban Flood Risk	• terrain model • river network layer • hydrological data on extreme rainfall events • meteorological data on extreme rainfall events • a land use map (different urban land uses) • Copernicus satellite imageries + VHR	Preprocessed	GB's	100s MB's	> 1	4	>=16
APP06 Urban Subsidence, Movements and Deformation Risk	• Ground deformations (Copernicus EGMS / Results from custom interferometric analysis - based on Sentinel-1 or VHR – prepared by GISAT before the EGMS is available) • Land use (Copernicus Urban atlas / Custom land use map) • Built-up expansion (WSF-Evo / GHSL)	Preprocessed	1s-10s GB's 1s-10s TB's (with MT InSAR chain)	10s-100s MB	> 1	1	16-32
APP07 Urban Air Quality	• CAMS: air quality reanalysis data • CAMS: air pollutant emissions • CLMS: Urban Atlas building height • OpenStreetMap: building locations and roadmap • C3S: 2m temperature and 10m (vectorial) wind speed • EIONET: Point Source emissions reported for LRTAP convention	Preprocessed	30 GB	200MB	> 1	122	16

APP08 Thermal Comfort	• UrbClim meteorological variables (100 m) • VHR land use map	Preprocessed	GB	100s MB’s	> 1	1	>=16
APP 09 Urban Heat Storage Monitoring	• Dynamin surface temperature (from App01) • Dynamic Surface Cover Fractions (100m) • 3D information on the buildings and trees height • Net-all wave radiation flux • Time rate of change for net all-wave radiation at the surface • Coefficients associated to response of the surface cover due the energy input	Preprocessed	0	100s MB’s	1		
APP10 Nature-based Solutions	• LiDAR, • Euro Maps 3D • DSM generated by DLR (tri-stereo imagery) • NDVI • Imperviousness • Albedo • Cadaster information (year of construction and protection level)	Preprocessed	10s GB’s	100s MB’s	> 1		8
APP11 Health impacts	• Air quality data from Sofia	Preprocessed					

3.2 System Requirements Summary

3.2.1 Data Requirements

A lot of different input data is required by the different Apps. Although all DIAS platforms contain a large number of Copernicus data, several sources need to get uploaded. After deciding on which DIAS the services will run and depending on its set of Copernicus data, WP2 has to guarantee the access to this data via the Copernicus Core Service Interface (CCSI).

The main requirement therefore is that the System can integrate the CCSI from WP2.

A lot of this input data is preprocessed, this must be considered in the system design. This can be done as preprocessing tasks that are part of the service workflows or the data can get preprocessed beforehand and then made available to the apps.

Therefore, the ability to preprocess the data is an additional requirement for the system.

3.2.2 Hardware Requirements

The provided file sizes will be helpful for setting up a sufficient storage.

The hardware requirements (CPU's and RAM) were not filled out for some of the services. The reason for this is, that these services are still under development. Without this information the virtual machines, which will host the services, might not be able to run all services successfully whereas memory errors might occur. To avoid this, the system must be designed in a way that, according to the final requirements of the services, additional Hardware sources can be made available.

3.2.3 Software Requirements

• To share and version code, a distributed version-control system is needed. Every service should be contained in a repository. The repository must only contain code, tests and docker instructions.

• To deploy the service, it must be dockerized. The dockerization of the service allows it to run on any machine independently. Optionally a deploy pipeline can fasten up the process of deployment.

• To instruct the service with the right parameters, an execution command must be provided. Additionally, the input data that is needed for the service to run must be provided. This data can be either raw data or preprocessed data. If the service consists of more than one part (Docker Container), the order and parallelism need to be given.

• To start the service via the API, a list of parameters must be given. In order to ensure that the information (input parameters) from the user is correct, one REGEX for each parameter needs to be provided. Furthermore, to ensure the quality of code, maintainability and fail safeness some implementations to the code of the Service are mandatory.

• To maintain the code, logging should be used. GeoVille provides an API and a python library to log to the central logging system. The results will be stored in a database.

3.2.4 General Requirements

To guarantee a fully functional system, the system must provide at least these functionalities:

- Authentication: Get access to the System infrastructure
- Service Submission: Submit a new service (order)
- Service Monitoring: Get the order status of the submitted service
- Retrieve Results: Download the result of the service order

4 CONCLUSION

The aim of this deliverable is to summarize all information's necessary, which are needed for the next step, the CURE System design.

Although there are some open questions regarding the final Hardware Requirements, it delivers all information's needed regarding the needed functionalities.

As the Services, which will be using the system, are still in development, an agile based development of the system will be necessary to adopt to new or changing requirements.

According to the given information, the most suitable DIAS will be evaluated. The system will then be designed in the next step of the project, in preparation for the transformation and deployment on the DIAS infrastructure.

5 APPENDICES

APPENDIX A: QUESTIONNAIRES TO SERVICE PROVIDERS

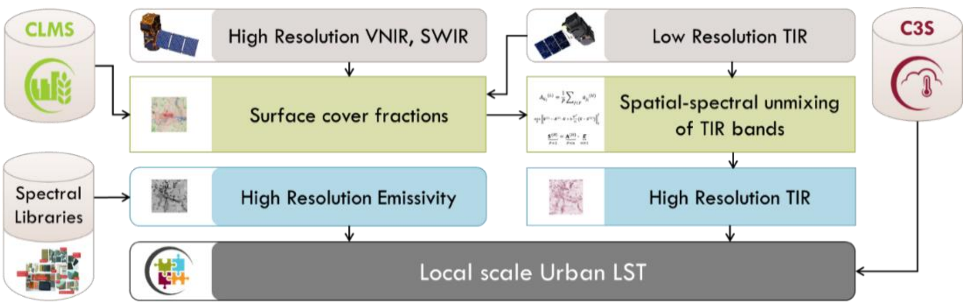
This questionnaire is an update on the one presented earlier in Deliverable D6.1.

Ap01 Local scale surface temperature dynamics

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.

The CURE Local Scale Surface Temperature Dynamics App to be developed by FORTH, is aiming at time series of urban surface temperature maps in local scale based on Copernicus Data. The surface temperature is one of the most important parameters in the physical processes of urban surface energy, water balance and the land-atmosphere exchanges. In this context, this CURE application will be utilized in all urban areas involved in the project, leading to frequent local scale surface temperature estimations, which are essential data for other CURE applications, i.e. the Surface Urban Heat Island (UHI) Assessment application, the Urban Heat Emissions Monitoring application and the Urban Heat Storage Monitoring application.

The surface temperature retrieval will be based on a downscaling algorithm (Mitraka et al., 2015[1]) which downscales Copernicus Sentinel-3 thermal data of 1 km spatial resolution to 100 m spatial resolution, using surface cover information of higher spatial resolution. In CURE the surface cover information will be retrieved from Sentinel-2 and combined CLMS information. Ancillary information from spectral libraries is used for the estimation of urban emissivity and atmospheric information from C3S for the atmospheric correction in the thermal infrared.



[1] Mitraka, Z., Chrysoulakis, N., Doxani, G., Del Frate, F., Berger, M., 2015. Urban Surface Temperature Time Series Estimation at the Local Scale by Spatial-Spectral Unmixing of Satellite Observations. Remote Sens. 7, 4139–4156. https://doi.org/10.3390/rs70404139

- **What data does your service need?**

– Surface Cover in 2 m spatial resolution (Very High Resolution Land Cover products)
– Red and InfraRed (NIR) measurements (10m Sentinel-2)

- Thermal infrared (TIR) measurements (1 km) from Sentinel-3
- Atmospheric information (water vapour) matching the TIR satellite acquisition time.

- **Does your code run with raw input data or does it require preprocessed data?**

  The code does not run with raw Copernicus input data, the data need to be preprocessed.

  Specifically:

- Sentinel-2 Data must be unzipped
- Sentinel-3 Data must be reprojected to WGS 84 UTM xxN and band clipped leaving 3 bands: Cloud.in, S8.in and S9.in. The data must be in .tif format.

- Water Vapour Data must be reprojected to WGS 84 UTM xxN in .tif format. The .tif file must contains 8760 bands (365 days x 24hours) **How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)**

  For each city: interim data ~ 3TB and output data 200MB.

  As output data the code produces a number of Geotiff files with a typical size in the order of a few hundred Mb.

- **Does your service contain more than one task (series of script executions)?**

  I is a single script with several functions. The preprocessing may be done separately to make interim products to other apps.

- **What resources does your service need for processing one unit (Cores, RAM)?**

  The codes can run on a single core with 4 (or more) GB RAM and runs for 1 to 3 hours locally for a year of each city (Disk 3000MB/s, Single Core 3.8GHz, RAM 3200MHz). The code can run in multi-tasking identifying from the beginning the number of cores, decreasing the running time. Each extra core needs independently 3GB RAM.

- **Do you use Docker?**

  No, we will use Docker for the first time in CURE.

Question 2: Will your service be user-interactive (how?) or do you only provide output data/maps?

It will only provide output urban surface temperature maps.

We can definitely discuss this with the users, but it is highly unlikely that they want to adjust the model parameters. It would require specific scientific knowledge.

Question 3: How do you see your service evolve on the long term (after the CURE project)?

We aim to continue this work after the CURE project and integrate the service in the Copernicus framework. The urban temperature service can be used in each of these cities to make local scale surface temperature time series maps.

Question 4: What is the added value for scientists/urban stakeholders over existing services?

- Is there any commercial service available in the market that provides a similar solution? (i.e. which are the main "competitor services" offering alternative solutions to yours)? (Note: please list the "applications")
- What are their strengths and weaknesses comparing to your service?
- Do you know any new service in progress of development (with TRL 7 or 8) that could compete in the future with your service?

Land surface temperature (LST) products are available from the CLMS[1] but in low spatial resolution (~5 km) which makes them inappropriate for use for urban sites. Similarly, LST products are available in high spatial resolution from satellite missions like Landsat[2] and ASTER[3], but these provide low temporal resolution.

This CURE application will provide dynamic (high temporal resolution) local scale surface temperature (of high spatial resolution, 100 m) maps, with algorithms designed for the retrieval of urban temperature (accounting for the variability of urban materials).

- Strengths of existing services compared to the CURE service:
    - Operational algorithms easily applicable globally
- Weaknesses of existing services compared to the CURE service:
    - Algorithms designed for retrieval of urban surface temperature, accounting for the variability of the urban materials
    - Local scale estimations of urban surface temperature
    - Capturing the diurnal variation of urban surface temperature in local scale

There are several publications and scientific projects towards the direction (e.g. Zhang et al., 2013[4]), but no services available from those.

[1] https://land.copernicus.eu/global/products/lst

[2] http://rslab.gr/downloads_LandsatLST.html

[3] https://lpdaac.usgs.gov/products/ast_08v003/

[4] Zhan, W., Chen, Y., Zhou, J., Wang, J., Liu, W., Voogt, J., Zhu, X., Quan, J., Li, J., 2013. Disaggregation of remotely sensed land surface temperature: Literature survey, taxonomy, issues, and caveats. Remote Sens. Environ. 131, 119–139. https://doi.org/10.1016/j.rse.2012.12.014

Question 5: How easily can your service be upscaled to other cities (both in Europe and outside Europe)?

The method is transferable to every city, with the limitation of spectral library data for representative emissivity values and given that detailed land cover maps are available. For European cities where Urban Atlas data are available the method is easily transferable, assuming that the London spectral library data of urban materials is appropriate for the rest of European cities. With specific assumptions, or a more advanced approach on emissivity retrieval for cities worldwide, the service could operate on a global scale.

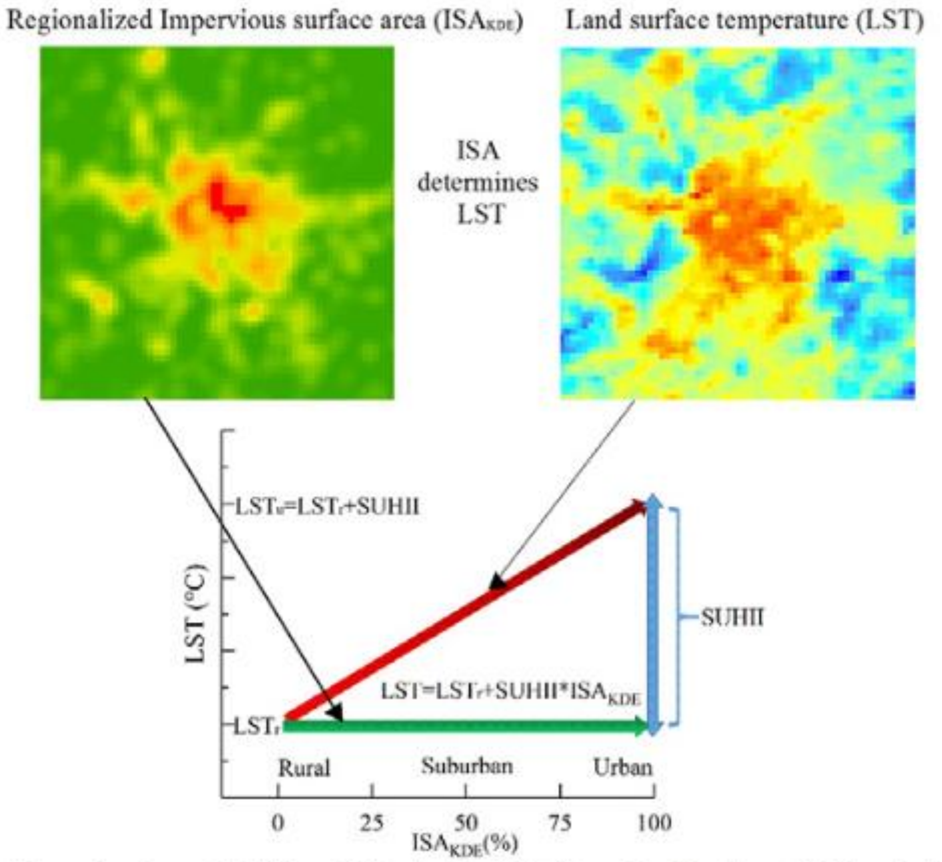
Question 6: Is your service linked with existing Copernicus services?

Yes, C3S and CLMS => see scheme above

Ap02 Surface Urban Heat Island Assessment

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.

App02 aims at estimating the Surface Urban Heat Island Intensity (SUHII) by exploiting (and possibly further improving) the methodology recently presented by Li et al. (2018). In particular, the intended approach will allow to calculate SUHII (and its temporal dynamics) by exploiting the linear relationship between Land Surface Temperature (LST) and the percentage impervious surface (PIS), which – according to the literature - proved consistent for cities in biomes dominated by forests and grasslands as in Europe.



Linear functions of LST is well fitted using ISA that regionalized by a KDE method (ISA$_{KDE}$). Slope of the functions is defined as the surface urban heat island ntensity.

- What data does your service need?

  The service requires as input the CLMS HR Imperviousness layers (IMD), as well as the Land Surface Temperature product generated in the framework of App01 (by exploiting both CLMS and CAMS products as input) both in GeoTiff format (the former at 10m resolution; the latter at 100m resolution).

Alternatively/complementary to the above, we are testing the effectiveness of employing an improved and updated imperviousness product derived from multitemporal Sentinel-2 imagery. In that case, one would need to gather all Level-2A Surface Reflectance S2 scenes with nominal cloud coverage lower than 60% intersecting the given area of interest in the entire target year and compute the maximum temporal NDVI for each pixel.

- Does your code run with raw input data or does it require preprocessed data?

  Both.

- How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)

  The envisaged output is a single number, or a time-series thereof, quantifying the SUHII for the given city of interest in the given time-frame and as such is in the range of few MBs. No spatial output is envisaged at the moment, however, pending further investigation a spatial display of model residuals could be made available. In this case the output data requirements scale with the area investigated, but should fall well within 10GB. The interim storage varies with the size of the targeted area of interest and the input data sources, but should likely not exceed 10GB in the case where S2 data have to be processed.

- Does your service contain more than one task (series of script executions)?

  This is still under development, but highly likely yes if, ultimately, we rely on an existing open software for part of the processing (e.g., simple geometrical operations with gdal, linear or empirical regression using the libsvm or pksvm tools) and own python scripts for the remaining.

- What resources does your service need for processing one unit (Cores, RAM)?

  Unfortunately, we cannot be very precise at this time here. Nevertheless, we envisage the whole process to be particularly fast and light; indeed, we are dealing with relatively small-size input GeoTiff images and a series of simple tasks to be executed. In the hypothesis that the maximum temporal NDVI has to be computed out of S2 imagery, the processing will become (slightly) more intense, with a higher RAM use, but likely well within 16GB. In this case the number of cores would depend on the performance requirements by the user. A reasonable performance should be expected with 4 to 8 cores.

- Do you use Docker?

  Yes.

Question 2: Will your service be user-interactive (how?) or do you only provide output data/maps?

Yes, optionally we would like the users to have the possibility of defining their specific area of interest for which to estimate the corresponding SUHII. This should happen ideally by drawing a polygon or uploading a vector geometry. Additionally, the users should also specify the time frame for which to perform the analysis.

Question 3: How do you see your service evolve on the long term (after the CURE project)?

Ideally, we foresee it to become the reference service for providing SUHII information for any city at European scale. Users should be able to check and compare the current and past SUHII for predefined cities (e.g., the 700+ included in the Urban Atlas), but also to request the computation for specific AOIs at specific times (provided the availability of the App01 LST product).

Question 4: What is the added value for scientists/urban stakeholders over existing services?

- Is there any commercial service available in the market that provides a similar solution? (i.e. which are the main "competitor services" offering alternative solutions to yours)? (Note: please list the “applications”)

  To our knowledge there is not yet any existing comparable service available.

- What are their strengths and weaknesses comparing to your service?

  /

- Do you know any new service in progress of development (with TRL 7 or 8) that could compete in the future with your service?

  No

Question 5: How easily can your service be upscaled to other cities (both in Europe and outside Europe)?

Upscaling should be straightforward, in the light of the full automation, the limited size of the input data and the envisaged light and fast processing. Given the availability of the HR imperviousness layer, for Europe it will be easier. Outside Europe is feasible, but shall rely then on the S2-based imperviousness only.

Question 6: Is your service linked with existing Copernicus services?

Yes. It is linked to the CLMS service directly and CAMS indirectly through the LST product from App01

Ap03 Urban Heat Emissions Monitoring

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.

The CURE Urban Heat Emissions App uses the aerodynamic resistance method for calculating sensible heat fluxes as described in Feigenwinter et al. (2018) [1] and CURE Deliverable D1.3 “Methodology review and selection”. A DOM (Digital Object Model) for buildings and trees, Land Surface Temperature and basic meteorological data (Air Temperature, wind velocity, wind direction, air pressure) is needed as input. The output is a map (GeoTiff, 100m resolution) with sensible heat flux values. The output is evaluated with in-situ flux tower data.

[1] FEIGENWINTER, C., VOGT, R., PARLOW, E., LINDBERG, F., MARCONCINI, M., FRATE, F. D. & CHRYSOULAKIS, N. 2018. Spatial Distribution of Sensible and Latent Heat Flux in the City of Basel (Switzerland). IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 11, 2717-2723.

- **What data does your service need?**

  LST from App01, high resolution LULC map, high resolution DOMs (trees and buildings), standard meteorology (air temperature, humidity, wind, wind direction, pressure, radiation). Flux tower data needed for validation

- **Does your code run with raw input data or does it require preprocessed data?**

  The code does not run with raw Copernicus input data, the data need to be preprocessed

- **How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)**

  Depending on time period and available scenes for LST (App01). One scene covers the previously defined AOI (area of interest) for all case study cities. In the case of Heraklion this is about 85 km^2 and one map needs 36 kB of storage capacity.

- **Does your service contain more than one task (series of script executions)?**

  No, it consists of one script.

- **What resources does your service need for processing one unit (Cores, RAM)?**

  tbd

- **Do you use Docker?**

  No

Question 2: Will your service be user-interactive (how?) or do you only provide output data/maps?

The service only provides output data/maps (GeoTiff)

Question 3: How do you see your service evolve on the long term (after the CURE project)?

Possible transferability/generalization to other cities provided the needed input data is available

Question 4: What is the added value for scientists/urban stakeholders over existing services?
Urban heat emission refers to the turbulent sensible heat flux, i.e. the heat exchange between the urban surface and the atmosphere. The sensible heat flux defines the amount of energy that is available for heating the urban atmosphere and it is thus closely related to air temperature. This amount of energy is strongly modified by the properties of the surface (land cover/ land use, 3D geometry) and the input of heat by human activities (traffic, buildings, industry). The localization of hot spots of high heat emissions will help urban planners to optimize their adaption strategies also with regard to heat stress, urban green space and building development.

- Is there any commercial service available in the market that provides a similar solution? (i.e. which are the main "competitor services" offering alternative solutions to yours)? (Note: please list the “applications”)

  There are commercial software packages (e.g. ENVI-met) that can deliver similar maps of sensible heat flux.

- What are their strengths and weaknesses comparing to your service?

  App03 only provides sensible heat fluxes (easy implementation, low cost), while commercial packages may process complete climate simulation (complex implementation and high costs).

- Do you know any new service in progress of development (with TRL 7 or 8) that could compete in the future with your service?

  No

Question 5: How easily can your service be upscaled to other cities (both in Europe and outside Europe)?

See question 3: Upscaling to other cities depends on the availability of input data.

Question 6: Is your service linked with existing Copernicus services?

CLMS (Urban Atlas and Building heights) if available
C3S Reanalysis data (alternative to in-situ meteorological data as input)

Ap04 Urban CO_2 emissions monitoring

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.

The CURE Urban CO_2 Emissions App uses a land use regression model (LUR) based on source area weighted and source strength weighted CO_2 fluxes from an urban flux tower for estimating city-wide CO_2 emissions. A detailed high resolution Land Use/Land Cover map, a DOM (Digital Object Model) for buildings and trees, basic meteorological data (Air Temperature, wind velocity, wind direction, air pressure) and profiles of source strengths derived from population density, traffic, vegetation dynamics, etc. are needed as input. The output is a map (GeoTiff, 100m resolution) with CO_2 emissions (anthropogenic and biogenic).

- **What data does your service need?**

  High resolution LULC map, high resolution DOMs (trees and buildings), standard meteorology (air temperature, humidity, wind, wind direction, pressure, radiation). Profiles (diurnal/seasonal courses) of source strengths derived from population density, traffic, vegetation dynamics, etc. Flux tower data and a footprint climatology is needed for establishing the regression model and evaluation.

- **Does your code run with raw input data or does it require preprocessed data?**

  The data need to be preprocessed

- **How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)**

  Depending on time period. One scene covers the previously defined AOI (area of interest) for all case study cities. In the case of Heraklion this is about 85 km^2 and one map needs 36 kB of storage capacity. To document CO2 emissions for one year (total and seasonal mean diurnal courses) a storage capacity of about 1 GB is estimated.

- **Does your service contain more than one task (series of script executions)?**

  tbd

- **What resources does your service need for processing one unit (Cores, RAM)?**

  tbd

- **Do you use Docker?**

  No

Question 2: Will your service be user-interactive (how?) or do you only provide output data/maps?

Output data/maps (GeoTiff)

Question 3: How do you see your service evolve on the long term (after the CURE project)?

The service will be in any case a prototype which has to be further developed. Transferability to other cities may be given in a later stage provided the needed input data is available (flux tower needed!)

Question 4: What is the added value for scientists/urban stakeholders over existing services?
The total urban CO_2 emissions have a spatial dimension due to the heterogeneous nature of urban land use/land cover and urbanization. In this CURE application, the CO_2 emissions are partitioned into an anthropogenic (traffic, heating/cooling) and a biogenic component (urban green space). Spatial planning strategies have an influence on the urban form, and consequently affect CO_2 emissions through changes in traffic patterns, energy consumption and location and extent of urban green areas. Knowing the portion of the anthropogenic and the biogenic part of CO_2 emissions in a high spatial resolution (neighborhood scale) will provide urban planners with an additional decision support tool for developing emission reduction strategies.

- **Is there any commercial service available in the market that provides a similar solution? (i.e. which are the main "competitor services" offering alternative solutions to yours)? (Note: please list the "applications")**

  Not to our knowledge

- **What are their strengths and weaknesses comparing to your service?**

  /

- **Do you know any new service in progress of development (with TRL 7 or 8) that could compete in the future with your service?**

  No.

Question 5: How easily can your service be upscaled to other cities (both in Europe and outside Europe)?

See question 3: The application will not be mature enough to be upscaled to other cities.

Question 6: Is your service linked with existing Copernicus services?

CLMS (Urban Atlas) if available
CAMS (GHG fluxes) for validation
C3S (in-situ observations) meteorological data

Ap05 Urban Flood Risk

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.

The CURE **AP05 - Urban Flood Risk** modelling service will combine hydrological modelling approaches, satellite data and overlay analysis to obtain information about exposure of urban features to potential floods.

Flood hazard exposure is achieved by combination of various means:

(i) flood inundation extent and frequency from historical EO data, (ii) geomorphic analysis using DEM/DSM for estimation of flood susceptibility, (iii) modelling inundation extent and depth using runoff and discharge patterns, morphology and land cover.

There are few options for hydrological modelling itself and estimation of magnitude of floods. All of them are based on open source codes SW packages (with different types of licensing). The most suitable solution for CURE 05 app will be selected based on user requirements and technical specifications. In previous implementations, GISAT used the Hec-RAS (US Army Hydrologic Engineering Center) or Flo-2D (US Federal Emergency Management Agency (FEMA) supported) SW suites for hydrological modelling. However, also the Anuga and OpenFOAM will be considered to be used for this purposes. All these models are based on combination of local terrain morphology and hydrological characteristics with intensity of precipitation and runoff during extreme rainfall events. Based on this, the expected magnitude of flood (extent, inundation depth) is modelled for scenarios of flood events with different magnitudes.

Additionally, this modelling approach based on hydrological models will be complemented by an approach based on Copernicus satellite data, taking into consideration flood events in the past and estimating flood probability (susceptibility) for each location inside the area of interest.

The main result of the modelling will be the maps showing flood extent, inundation depth or flood susceptibility for different magnitude of potential flood events (represented by specific intensity of rainfall and runoff). Additionally, flood risk to urban features/community assets will be estimated, combining the results of flood scenario modelling with the information about geographical distribution of urban features, coming from land use maps (Copernicus Urban Atlas or local urban planning maps). Flood hazard to population can be also modelled, based on known or estimated (e.g. via disaggregation) distribution of dwellers in the city.

Last, but not least, also the rapid flood monitoring information during recent or on-going events can be offered in the frame of this service, if requested by the user.

The results can be either shared as GIS layers in raster or vector format, which can be easily integrated into GIS platform of the city or in a form of maps with proper symbology and legend. However, we prefer to visualize results in an interactive web-based application, allowing the user to interactively explore and analyze the flood hazard for his city, and in combination with additional relevant data (DTM, land use maps).

- **What data does your service need?**

  The service needs detailed terrain model, river network layer and hydrological and meteorological data on extreme rainfall events. Moreover, a land use map is needed, as detailed as possible, distinguishing between different classes of urban land use. The second will be represented by Copernicus Urban Atlas by default and can be complemented by local urban planning maps if available. For the flood analysis based on satellite data, the Copernicus satellite imageries will be used as an input, which can be complemented by additional third-party VHR data for specific dates during or after flood events.

- **Does your code run with raw input data or does it require preprocessed data?**

  The code does not run with raw Copernicus input data, the data need to be preprocessed.

- **How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)**

  For each city:

  As interim data, the results of hydrological modelling are produced. The size depends largely on the chosen domain size, but is typically in the order of a few Gb.

  As output data the code produces one or more vector layers files with a typical size in the order of a few hundred Mb.

- **Does your service contain more than one task (series of script executions)?**

  Yes it is processing chain composed of several independent SW products.

- **What resources does your service need for processing one unit (Cores, RAM)?**

  The used SW products runs on a four core CPU with 16 (or more) Gb RAM and runs for a few hours to a day (depending on the domain size).

- **Do you use Docker?**

  GISAT has expertise with dockerization of applications. For this particular app, the feasibility of dockerization needs to be assessed, because of the dependencies on some proprietary SW packages, which, however, may be replaced by other solutions during the implementation process.

Question 2: Will your service be user-interactive (how?) or do you only provide output data/maps?

Yes, service will be user-interactive. User will alter inputs and modelling parameters. This would allow the user to model assess their own flooding scenarios.

Question 3: How do you see your service evolve on the long term (after the CURE project)?

The aim is to develop an application which will be livable after the CURE project ends and will work in an interactive and automated way, allowing the users to add new cities or modify input parameters and model different flood scenarios. This, however, may require some additional efforts still needed after the CURE project is finished, including development of a web-based platform and integration of advanced functionalities allowing user-driven parametrization of the modelling process.

In any case, our aim is to offer an application which will be robust in sense of applicability on any city with appropriate input data, and which can be highly beneficial for urban and city-regional planners and environmental departments of the cites and to offer this app to a wide range of end-users in the future.

Question 4: What is the added value for scientists/urban stakeholders over existing services?

- Is there any commercial service available in the market that provides a similar solution? (i.e. which are the main "competitor services" offering alternative solutions to yours)? (Note: please list the "applications")
- What are their strengths and weaknesses comparing to your service?
- Do you know any new service in progress of development (with TRL 7 or 8) that could compete in the future with your service?

For the competition, there are few well established similar commercial solutions and services, usually based on hydrological modelling and sure many new services in progress of development. For example, the OpenFlows FLOOD by BENTLEY[1], Flood Modeller by Jacobs[2], MIKE by DHI[3] or services provided by Waterman Group[4].

- Strengths of competitive solutions compared to our service:
  - Well-known in the consultancy business
  - Some of them allow for modelling in very high spatial detail
  - Some of them are taking into consideration also the sub-surface runoff supported by sewage drainage in the cities etc.
- Strengths of our service:

- Based on Copernicus data – allowing easy replication for any Copernicus city and also in the future, as Copernicus program guarantees harmonized data acquisition/production in following years
- Running on DIAS – no demands on user own processing infrastructure

Based on satellite imageries – allowing to perform both historical analysis and rapid monitoring as a response on recent events

[1] https://www.bentley.com/en/products/product-line/hydraulics-and-hydrology-software/openflows-flood

[2] https://www.floodmodeller.com

[3] https://www.mikepoweredbydhi.com/products/mike-flood/integrated-flood-modelling

[4] https://www.watermangroup.com/

Question 5: How easily can your service be upscaled to other cities (both in Europe and outside Europe)?

It should be very straightforward to upscale this service to other cities. The only limitation is the availability of the local hydrology parameter.

Question 6: Is your service linked with existing Copernicus services?

Yes, C3S (precipitation data) and CLMS (Urban Atlas), Copernicus satellite data (Sentinels and VHR satellites)

Ap06 Urban Subsidence, Movements and Deformation Risk

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.

The CURE Land Subsidence App aims to support cities with enriched information about land subsidence. To understand drivers and impacts of land subsidence at city level the intensity of terrain deformation hazard derived by means of MT InSAR technology needs to be integrated with additional supportive information. Analysis of spatial patterns of subsidence phenomena brings insights into hazard distribution and its magnitude. Especially important is identification of clusters of subsiding measurement points attributed to similar behavior and outlying points or clusters dissimilar to behavior of points in surroundings. Furthermore, given complex nature of potential drivers and size of observable objects in the urban environment multi-scale approach will be adopted. Distribution and chronology of urban evolution (expansion, densification) may affect both spatial and temporal patterns of land subsidence and should be assessed in relation to observed deformation rate. The module will consist of series of Python scripts responsible for spatial-temporal analytics and data fusion. Spatial associations will be analyzed within results from Copernicus European Ground Motion Service, EGMS (when it will be available), or from custom MT InSAR processing chain deployed on DIAS infrastructure. Next, the results will be integrated with temporal patterns derived from pan-European and global open datasets describing urban structure and height (Copernicus Urban Atlas with 3D component) and long-term built-up evolution (DLR's World Settlements Footprint-Evolution, JRC's GHSL). Output from module will be represented by standardized analytics providing added value derived from fusion of scalable inputs.

- **What data does your service need?**
  - Ground deformations (Copernicus EGMS / Results from custom interferometric analysis - based on Sentinel-1 or VHR – prepared by GISAT before the EGMS is available)
  - Land use (Copernicus Urban atlas / Custom land use map)
  - Built-up expansion (WSF-Evo / GHSL)
- **Does your code run with raw input data or does it require preprocessed data?**

  The code does not run with raw Copernicus input data, the data need to be preprocessed.

- **How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)**
  - Interim: 1s-10s GB (excluding custom MT InSAR chain); 1s-10s TB (including custom MT InSAR chain)
  - Output: 10s-100s MB
- **Does your service contain more than one task (series of script executions)?**

MT InSAR chain will be run independently. The analytical service will be modular, but it shall be run as one series of Python scripts

- What resources does your service need for processing one unit (Cores, RAM)?

  We cannot estimate this properly at this stage of the project. In general:

  MT InSAR chain: The more cores the better. The more RAM the better.

  Analytical chain 1 core, 16-32 GB RAM. Hours to days runtime.

- Do you use Docker?

  we will use Docker

Question 2: Will your service be user-interactive (how?) or do you only provide output data/maps?

Very limited interactivity is expected as the service is extremely asynchronous. Simple user interface shall allow definition of basic variables.

- Definition of the AOI
- Selection of input components to be fused (from the list)
  - Cop EGMS / custom MT InSAR outputs
  - Built-up evolution
  - Urban Atlas Land use map / Custom land use map

Specification of clustering options (variables)

Output maps shall be presented using proprietary web platform

Question 3: How do you see your service evolve on the long term (after the CURE project)?

We would like to continue running and offering the service after the CURE for more cities in Europe (primarily using Copernicus data: EGMS & Urban Atlas) or outside Europe (primarily using custom MT InSAR & land use data).

The service shall provide set of standardized derivative analytics related to spatial and temporal distribution of ground deformations based on fusion with Copernicus and global open datasets.

Question 4: What is the added value for scientists/urban stakeholders over existing services?

- Is there any commercial service available in the market that provides a similar solution? (i.e. which are the main "competitor services" offering alternative solutions to yours)? (Note: please list the "applications")
- What are their strengths and weaknesses comparing to your service?
- Do you know any new service in progress of development (with TRL 7 or 8) that could compete in the future with your service?

1. We do not know of competing equivalent commercial service. Several European companies are providing MT InSAR services, e.g.: TRE-Altamira, e-Geos, SkyGeo, Sensar, Sarsense, Gamma RS, Planetek, NHazka, and several other start-up companies and institutions (BRGM, KIT, NGO, …). EGMS online visualization platform will not provide comparable functionality.
2. Most of the MT InSAR providers provide their own online platform for exploration of complex InSAR results with likely similar basic functionalities and specialized functions and visualizations depending on each service provider's expertise and business model.
3. There might be other similar services in development. Gisat combines practical and scientific background in development of solutions based on MT InSAR technology, and proven record in deployment commercial online platform integrating and fusing MT InSAR results.

Question 5: How easily can your service be upscaled to other cities (both in Europe and outside Europe)?

It should be easy to upscale for cities in Europe where Urban Atlas land use map is available (after EGMS results from its first phase are operationally available in 2022). In the meantime, custom MT InSAR service need to be utilized to obtain comparable ground deformation data. Custom MT InSAR service might be generally utilized if customer needs input with different parameters, e.g. with higher spatial density of measurement points, custom period for deformation rate estimation, or using different deformation model.

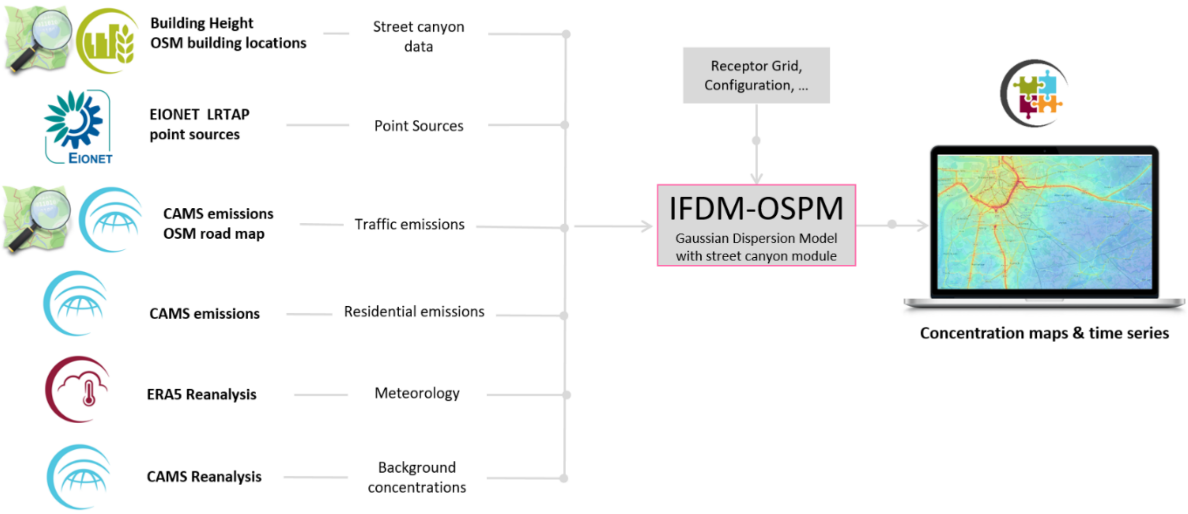
Outside Europe ground deformation component relies on execution of custom MT InSAR service chain. Scaling depends on availability and nomenclature of custom land use map. If not available at all, the service may be downscaled to provide analytics only using global open datasets (WSF-Evol, GHSL).

Question 6: Is your service linked with existing Copernicus services?

Yes. Urban Atlas and also EGMS (when ready).

Ap07 Urban Air Quality

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.



- What data does your service need?

  Minimal data:

  - CAMS: air quality reanalysis data
  - CAMS: air pollutant emissions
  - CLMS: Urban Atlas building height
  - OpenStreetMaps: building locations and roadmap
  - C3S: 2m temperature and 10m (vectorial) wind speed
  - EIONET: Point Source emissions reported for LRTAP convention
- Does your code run with raw input data or does it require preprocessed data?

  Preprocessed data (for all the datasets)

- How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)

  For one city:

  - Interim storage (temporary model output, which is later combined to annual mean concentration): 30 GB
  - Final output: 200MB
- Does your service contain more than one task (series of script executions)?

  Series of scripts: one Python preprocessing script per data input (point sources, traffic emissions, residential emissions, streetcanyon data, meteorology, background concentrations), scripts to start the model core (python wrapper around Fortran model core), scripts for postprocessing.

- What resources does your service need for processing one unit (Cores, RAM)?

  Depends on the required runtime.

  RAM: 16 GB

  Number of cores: the more the better (the annual mean concentration is modelled day by day, but all days can run in parallel). Typically, we use 122 cores. Results are then available in 24 hours.

- Do you use Docker?

  No.

Question 2: Will your service be user-interactive (how?) or do you only provide output data/maps?

Only output data. The model core is commercial software (partially owned by VITO, partially by Aarhus University). The AQ app is not scheduled to be implemented in the DIAS.

Question 3: How do you see your service evolve on the long term (after the CURE project)?

Continued updates based on model validations.

Question 4: What is the added value for scientists/urban stakeholders over existing services?

- Is there any commercial service available in the market that provides a similar solution? (i.e. which are the main "competitor services" offering alternative solutions to yours)? (Note: please list the “applications”)
- What are their strengths and weaknesses comparing to your service?
- Do you know any new service in progress of development (with TRL 7 or 8) that could compete in the future with your service?

We consider our service as a consultancy service, whereby high resolution urban pollutant concentration maps are provided as a service to regional / national / urban authorities, or to institutes providing pan-European data. Some more information on the these four type of clients:

- Many European national / regional or urban authorities (e.g. the UK, France, Belgium, the Netherlands, Germany, Sweden, Finland ….) have many year’s expertise in producing air quality maps using AQ models for their region and cities. More recently, at high resolution as we propose here. These models are often tailored to use the local datasets (e.g. traffic data, fleet composition), and therefore provide a much better quality than the CURE AQ application. In some cases the authorities run the models themselves, in other regions the work is often subcontracted out (as frameworks or contracts) to local experts: at universities, research institutes or consultancies. Bottomline: they prefer to user their own local datasets.

- The next group of stakeholders are cities / regions / countries who do not use a high-resolution AQ model but do have a lot of detailed information (traffic flows, fleet data, information on residential sources). For these stakeholders, the quality of an AQ application based on the local data will always outperform the AQ model solely based on Copernicus data. Thus again, they prefer to user their own local datasets.
- A third group of stakeholders are those cities / regions / countries lacking sufficient local datasets and resources (economically and technically) to prepare urban scale maps, to which we aim this CURE service.
- A final group of potential clients considers the institutes willing to provide a pan-European service. At these moment, there are no services that model the air quality at street-level scale throughout Europe. There are however services relying on modelling at a lower resolution, and services that rely on air quality measurements. Both types of services however often fail to incorporate the street canyon contributions[1], which are important for the population exposure. A pan-European high-resolution (street-level) modelling service would moreover provide the ideal tool to compare cities with each other.

Competitors:

- There are some competitors that offer a similar service. An example is ADMS Urban software (TRL9) (http://www.cerc.co.uk/environmental-software/ADMS-Urban-model.html#:~:text=What%20is%20ADMS%2DUrban%3F,urban%20areas%2C%20cities%20and%20towns.&text=investigation%20of%20air%20quality%20management,air%20pollution%20exposure%20studies), which provide a similar consultancy service. We have somewhat more experience with Copernicus data as driver for the models (important for a pan-European service), and we have more connections with the market in Eastern EU (important for services for urban, regional and national authorities in that region).
- Many open source air quality models exist (e.g. Aermod). These models could be used in combination with the CAMS data by local authorities or research institutes. It however requires a lot of experience with big data and (linux) programming to get these models up and running, leading to high costs.

For detailed information and competitors and the market situation: see reports of the EU H2020 AirQast project.

[1] For the low-resolution modelling chains, street-canyons or not considered in the chain. Services based on measurements, on the other hand, are strongly influenced by the location of the measurement stations. As many cities lack stations in street canyons (e.g. Sofia), these services will also miss the peak values.

Question 5: How easily can your service be upscaled to other cities (both in Europe and outside Europe)?

Current service can easily be deployed for all EU cities (although some assumptions will have to be made for buildings height for non-capitals, as the building height layer of CLMS is only available for the capitals).

Deployment outside Europe requires a lot of changes, as many of the currently used datasets are only available EU-wide (emissions, background concentrations). Often global alternatives exist (e.g. CAMS global emissions and reanalysis), but the quality of these product is often lower.

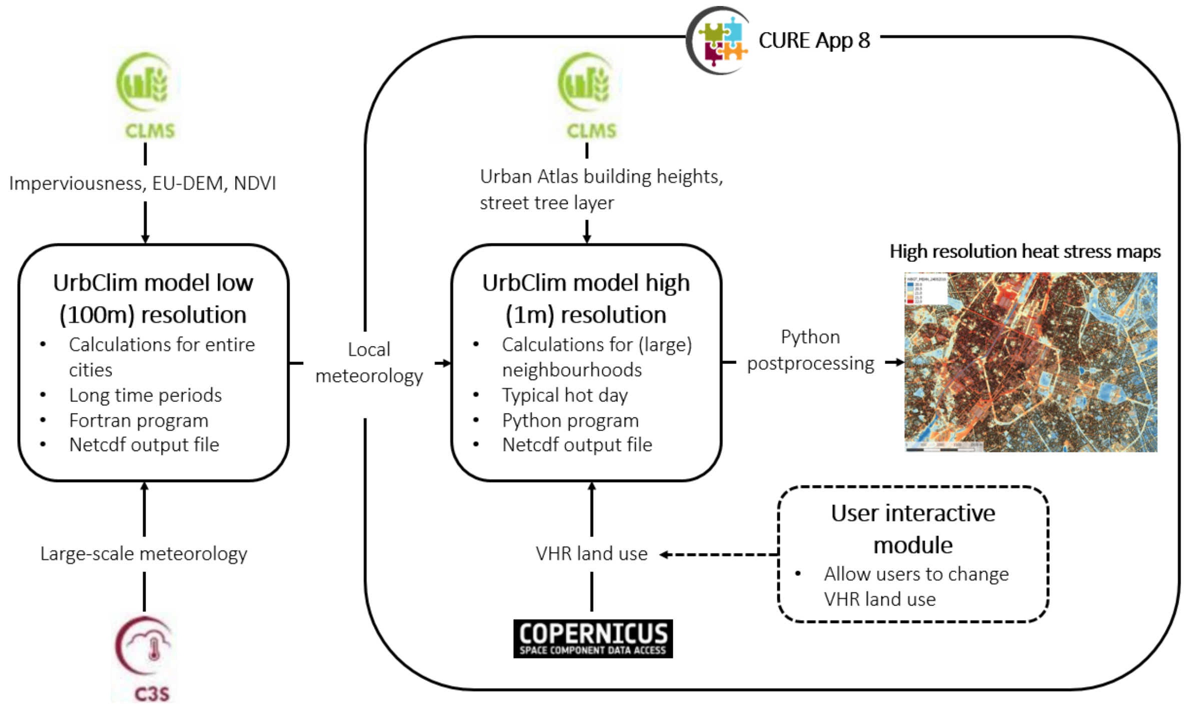
Question 6: Is your service linked with existing Copernicus services?

CAMS, C3S, CLMS => see data

Ap08 Urban Thermal Comfort

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.

The CURE Thermal Comfort App is based on the urban climate model UrbClim, developed by VITO, for which recently a high resolution thermal comfort module is developed (Lauwaet et al., 2020[1]). This module calculates hourly Wet Bulb Globe Temperature values for a selected typical hot day in summer (Netcdf file), which are translated in postprocessing to heat stress maps, based on international or local threshold values (Geotiff file). The module consists of a series of Python scripts. As input data, local meteorological variables (air temperature, humidity, wind speed, radiation and land surface temperature) are needed that are provided by the 100m resolution UrbClim model. Furthermore, a very high resolution land use map is needed, which is combined with building height and tree height information in a preprocessing step.



[1] Lauwaet D., Maiheu B., De Ridder K., Boënne W., Hooyberghs H., Demuzere M., Verdonck M.-L., 2020. A New Method to Assess Fine-Scale Outdoor Thermal Comfort for Urban Agglomerations. Climate, 8, 6; doi:10.3390/cli8010006.

- What data does your service need?

  The service needs 2 input files: a Netcdf file containing the 100m resolution UrbClim meteorological variables, and a Geotiff file with the very high resolution land use map.

The Netcdf files with meteorological variables (1 for each city) will be stored inside the Docker container as the users of the App don't need to change them. Only the geotiff file with the VHR land use map needs to be stored somewhere or collected from an external site outside the Docker container. This is needed to allow the users of the App to modify the land use map and run scenarios.

- Does your code run with raw input data or does it require preprocessed data?

  The code does not run with raw Copernicus input data, the data need to be preprocessed.

- How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)

  For each city:

  As interim data, the code produces a few Geotiff files and a Netcdf file. The size depends largely on the chosen domain size, but is typically in the order of a few Gb.

  As output data the code produces one or more Geotiff files with a typical size in the order of a few hundred Mb.

- Does your service contain more than one task (series of script executions)?

  No, it consists of one series of Python scripts.

- What resources does your service need for processing one unit (Cores, RAM)?

  The code runs on a single core with 16 (or more) Gb RAM and runs for a few hours to a day (depending on the domain size).

- Do you use Docker?

  Yes, we have expertise with Docker. VITO's IT team uses Docker regularly to run our models in a client's environment.

Question 2: Will your service be user-interactive (how?) or do you only provide output data/maps?

Ideally, we have a user-interactive service in mind, where the user can alter (or upload) the input very high resolution land use map. This would allow the user to assess their own urban adaptation scenarios.

This aspect of the service can perhaps be discussed/developed in collaboration with GISAT, who have experience with user-interactive map services.

Question 3: How do you see your service evolve on the long term (after the CURE project)?

We aim to continue this work after the CURE project and integrate the service in the Copernicus framework. As VITO has already delivered 100m resolution urban climate data for 100 European cities to Copernicus, our service can be used in each of these cities to make high resolution thermal comfort maps. The service could also be linked to CMIP5 (or CMIP6) climate change data, to assess whether green-blue urban adaptation measures are effective in reducing the heat stress impact of climate change.

Question 4: What is the added value for scientists/urban stakeholders over existing services?

- Is there any commercial service available in the market that provides a similar solution? (i.e. which are the main "competitor services" offering alternative solutions to yours)? (Note: please list the “applications”)
- What are their strengths and weaknesses comparing to your service?
- Do you know any new service in progress of development (with TRL 7 or 8) that could compete in the future with your service?

There are a few commercial software packages (e.g. ENVI-met[1] and FLUENT[2]) that can deliver similar high resolution thermal comfort maps for buildings/city quarters and that are used by consultancy firms/research institutes in projects for urban authorities.

Strengths compared to our service:

- Well-known in the consultancy business
- Allow for a lot of detail in the input data (lots of scenario options)

Weaknesses compared to our service:

- Expensive to set up and run
- Not possible to model large areas
- Make use of idealized settings, difficult to validate against measurements

Some consultancy companies/research institutes make use of high resolution remote sensing imagery, combined with empirical relations or measurements, to estimate spatial heat stress patterns in cities (e.g. Koopmans et al., 2020[3]).

Strengths compared to our service:

- Low-cost methodology
- Can be applied for very large areas

Weaknesses compared to our service:

- Not possible to do scenario analyses

- Difficult to validate against measurements

Most probably there are other services in development since this is a growing market, but none of these have the direct links to Copernicus and scientific background that our service has.

[1] https://www.envi-met.com/buy-now/

[2] https://www.ansys.com/products/fluids/ansys-fluent

[3] https://www.sciencedirect.com/science/article/pii/S0360132320303644

Question 5: How easily can your service be upscaled to other cities (both in Europe and outside Europe)?

It should be very straightforward to upscale this service to the 100 European cities that have been modelled with the UrbClim model in the past, and for which the data are available in the Copernicus Data Store. The only limitation is the availability of the VHR land use map, but we assume these are/will become available soon for a large number of cities, or can be made for a limited cost.

For other cities (both inside and outside Europe) we first need to perform 100m resolution UrbClim simulations, which comes with an additional cost.

In principle this service can relatively easy operate on a global scale

Question 6: Is your service linked with existing Copernicus services?

Yes, C3S and CLMS => see scheme above

Ap09 Urban Heat Storage Monitoring

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.

The CURE Urban Heat Storage Monitoring App to be developed by FORTH and UNIBAS, is aiming at delivering heat storage estimations over cities. Urban Heat Storage ( ) is the net flow of heat stored in urban canopy and represents all the mechanisms of storage of energy within the volume, i.e., the air, on trees, in buildings constructed in the ground, etc. Observations of global temperature evolution indicate a pronounced air temperature warming, since an increase in the occurrence of heat waves and the Urban Heat Island (UHI) effects tends to exacerbate such warming. Among all the effects caused by the substitution of natural ecosystems for urban land-use, the most pronounced is the increase in the amount of energy stored in the urban canopy (especially in buildings), which is approximately 2-6 times larger than in non-urban canopies. The slow release of this energy, stored mainly in the buildings and paved surfaces of cities, causes the UHI effect and it is therefore related to the energy efficiency and consumption in cities. In this framework, the CURE application will deploy various EO and in-situ urban data, such as land cover, geometry, radiation and air/surface temperature, towards monitoring urban heat storage.

The urban heat storage retrieval will be based on the Objective Hysteresis Model (Offerle et al. 2005[1]).

[1] Offerle, B., Grimmond, C.S.B.B., Fortuniak, K., 2005. Heat storage and anthropogenic heat flux in relation to the energy balance of a central European city centre. Int. J. Climatol. 25, 1405–1419. https://doi.org/10.1002/joc.1198

- **What data does your service need?**
  - Dynamin surface temperature (from AP01)
  - Dynamic Surface Cover Fractions in 100 m spatial resolution
  - 3D information on the buildings and trees height
  - Net-all wave radiation flux
  - Time rate of change for net all-wave radiation at the surface
  - Coefficients associated to response of the surface cover due the energy input (i.e. thermal properties of the urban materials in the city).
- **Does your code run with raw input data or does it require preprocessed data?**

  The method does not run with raw Copernicus input data, the data need to be preprocessed.

- **How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)**

  For each city:

The method will not probably not produce any interim data.

As output data the method will produce a Geotiff files with a typical size in the order of a few hundred Mb.

- Does your service contain more than one task (series of script executions)?

There is no service available at the moment. The scripting is under development, but we aim to build it into a single script with several functions.

- What resources does your service need for processing one unit (Cores, RAM)?

NA

- Do you use Docker?

No, we will use Docker for the first time in CURE.

Question 2: Will your service be user-interactive (how?) or do you only provide output data/maps?

It is highly unlikely that this application could be made user-interactive.

Question 3: How do you see your service evolve on the long term (after the CURE project)?

We aim to continue this work after the CURE project, but lots of work would necessary before this can be a service integrated in the Copernicus framework.

Question 4: What is the added value for scientists/urban stakeholders over existing services?

- Is there any commercial service available in the market that provides a similar solution? (i.e. which are the main "competitor services" offering alternative solutions to yours)? (Note: please list the “applications”)
- What are their strengths and weaknesses comparing to your service?
- Do you know any new service in progress of development (with TRL 7 or 8) that could compete in the future with your service?

There are no similar services available for delivering urban heat storage maps from satellite and other EO data. All efforts are at the moment on a research level and the CURE app will be the first to allow insights on the method transformation into a service.

Question 5: How easily can your service be upscaled to other cities (both in Europe and outside Europe)?

Very difficult. It would require data collection, manual interpretation and a lot of scientific experimenting, for transferring the method to other cities.

Question 6: Is your service linked with existing Copernicus services?

Yes, C3S and CLMS since it uses input data from AP01.

Ap10 Nature-based Solutions

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.

- **What data does your service need?**

  The NBS App will compare 3 different scenarios based on different resolution inputs: LiDAR, Euro Maps 3D, DSM generated by DLR (tri- stereo imagery), NDVI, imperviousness, albedo and cadaster information (year of construction and protection level). The NBS App sets its basis on the methodology proposed in Decumanus, which identifies potential green roofs from slope computation and identification of % flat surface. It also combines NDVI to detect surfaces already vegetated. The definition of flat or quasi-flat roofs will be adjusted on a case by case basis, according to local policies. Prioritization of potential green roof will be then calculated by discarding protected buildings where such interventions are not feasible due to legal restrictions, again considering a case by case approach. A threshold for load capacity based on the year of construction will determine which buildings are more suitable to host green roofs without major interventions on the structure. Albedo calculation and imperviousness will also be considered as factors to determine the areas where green roofs are prioritized. It is based on a series of Python scripts. The idea is that some functions will be interactive, and the users will have the possibility to select all parameters or choose only some of them.

  The nature of the application makes it suitable for "static assessment". Potential and prioritization of green roofs is evaluated once, in a precise time period and do not require short term evolution assessment of parameters.

- **Does your code run with raw input data or does it require preprocessed data?**

  Preprocessed data

- **How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)**

  Output should be around few hundreds of Megabytes, while interim would be much heavier (in the order of some tens of Gigabytes).

- **Does your service contain more than one task (series of script executions)?**

  Yes, it contains a series of scripts that run lineally

- **What resources does your service need for processing one unit (Cores, RAM)?**

At present we cannot say much for the number of cores. For the RAM, 8GB of RAM might be suitable.

- Do you use Docker?

  No

Question 2: Will your service be user-interactive (how?) or do you only provide output data/maps?

Main output is a map but in the final phase of the process may include user interactive component

Question 3: How do you see your service evolve on the long term (after the CURE project)?

Evolution in the long-term may include an inventory of solutions to help choosing the appropriate type of green roof or complementary solutions according to effectiveness parameters and building constructive typology.

Question 4: What is the added value for scientists/urban stakeholders over existing services?

- Is there any commercial service available in the market that provides a similar solution? (i.e. which are the main "competitor services" offering alternative solutions to yours)? (Note: please list the “applications”)

  Green City is an online tool developed by TerraNIS for the study, monitoring and management of urban vegetation and green areas in cities and to accompany them in their ecological transition. It provides a diagnosis of the vegetation, quantifies it and qualifies it considering biodiversity and well-being aspects.

  For that the tool uses satellite images of different resolutions (Pléiades and Sentinel-2 images), open data (Open street map and other sources) and INSEE data (French National Institute for Statistics and Economic Studies). The processing of this data allows the calculation of different simple and synthetic indicators that can be displayed in a dashboard. It also allows to guide decisions and select priority areas of action, by introducing target values for each indicator.

  Despite not being a tool focused on green roofs, in their diagnosis green roofs could also be included. Besides this, the tool includes the possibility to prioritize intervention areas thanks to the possibility of introducing target values for the indicators, that is something that could be included in our development.

- **What are their strengths and weaknesses comparing to your service?**

  Our tool is more specific, and it is focused on the greening potential of roofs, excluding those that could not been transform into green roofs for different reasons (slope, load limits, etc.).

  City Green is focused in the diagnosis of the green areas of the city (that could also include green roofs, but not sure about this), but does not calculate its greening potential. Introducing target values the user can identify the areas that don't reach the values in order to prioritize the interventions, but this does not mean that reaching the objective will be feasible.

  The service is only available for Toulouse it is necessary to contact them to include new cities

- **Do you know any new service in progress of development (with TRL 7 or 8) that could compete in the future with your service?**

  A paper of 2016 on estimating potential green cover at rooftop level using 3D data obtained by LiDAR was analyzed, but seems that funds were linked to national and post-doctoral grant and no advances have been found on the method used. *Santos T et al. (2016).Quantifying the City's Green Area Potential Gain Using Remote Sensing Data. Sustainability 2016, 8(12), 1247; https://doi.org/10.3390/su8121247*

Question 5: How easily can your service be upscaled to other cities (both in Europe and outside Europe)?

It depends on the availability of inputs data

Question 6: Is your service linked with existing Copernicus services?

Yes, mainly CLSM, but can be associated to C3S and thermal comfort application (to be further discussed).

Ap11 Health Impacts

Question 1: Describe your service-to-be in technical detail (include schemes if possible), while addressing the sub-questions.

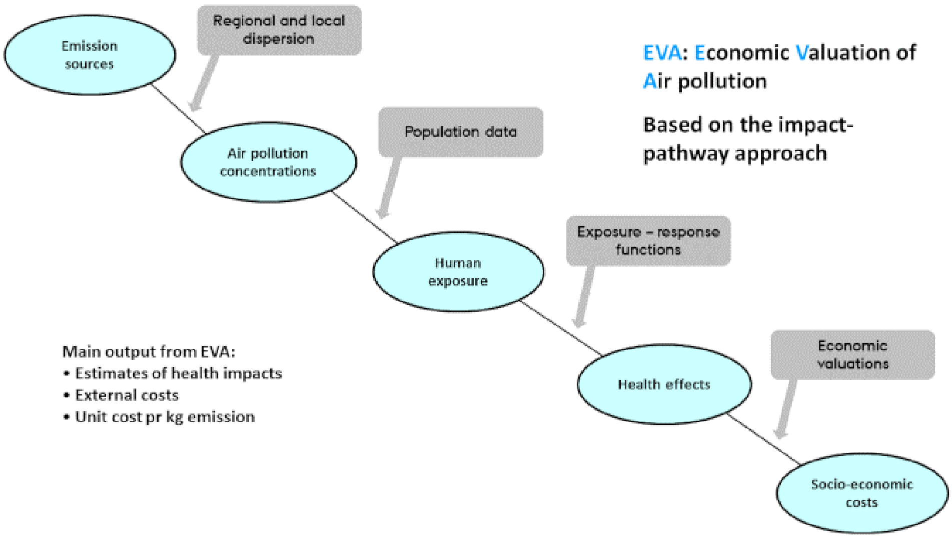
Application 11 Health impacts and health costs of air pollution will build on the existing EVA model developed by Aarhus University. The integrated model system is based on the impact-pathway chain and is used for assessment of health impacts from air pollution, including both health effects and related external costs (sometimes also referred to as "indirect costs") which can be attributed to air pollution exposure. Chemical components important for health impacts and included in the EVA system are: nitrogen dioxide (NO_2), sulphur dioxide (SO_2), ozone (O_3) and particulate matter ($PM_{2.5}$), where the individual constituents of $PM_{2.5}$ are: mineral dust, black carbon (BC), organic matter (OM), secondary inorganic aerosols (SIA – i.e. nitrate, sulphate and ammonia), secondary organic aerosols (SOA) and sea salt.

The EVA model is coupled to the air pollution models DEHM (Danish Eulerian Hemispheric Model) and UBM (Urban Background Model) for regional-scale and local-scale health impact assessments, respectively. EVA includes gridded population data, exposure-response functions for health impacts in terms of morbidity and mortality, and economic valuation of the health impacts from air pollution.

The EVA system includes the best available and most accurate, yet computationally demanding methods, used in each part of the impact-pathway chain. The EVA system uses comprehensive and thoroughly tested chemistry-transport models when calculating air pollution levels in general as well as scenarios describing how specific changes in emissions of pollutants to the air affect air pollution levels at regional and local scale.

To calculate the impacts of the total air pollution levels or of emissions from a specific source or sector, concentrations and address-level population data are combined to estimate human exposure, and then the response is calculated using an exposure-response function (ERF) of the following form: $R = A \cdot C \cdot P$, where R is the response (e.g. in cases, days, or episodes), C the concentration (i.e. the total concentration or additional concentration resulting from emissions of a particular emission source), P the affected share of the population, and A is an empirically determined constant for the particular health outcome, typically obtained from published cohort studies.

We are not sure yet how the model runs in more technical detail than this, as it is patented and copyrighted by Aarhus University and will only be made available to CWare/CURE as an output analyses. However the service will be city specific in the beginning, before we can hopefully upscale the service to include more cities. However, if the air quality data is in the same format and year, for example for Berlin and Sofia, the model would be able to run both cities at once and provide output data for both Berlin and Sofia. But the analyses of the results would require a bit more time when two cities are included, compared to just one city.



- What data does your service need?

  Air quality data for Sofia from VITO

- Does your code run with raw input data or does it require preprocessed data?

  Processed data, format to be agreed with VITO

- How much storage does your interim and output data need? (Please provide two sizes, one for interim and one for output data)

  tbd

- Does your service contain more than one task (series of script executions)?

  tbd

- What resources does your service need for processing one unit (Cores, RAM)?

  tbd

- Do you use Docker?

  Yes

Question 2: Will your service be user-interactive (how?) or do you only provide output data/maps?

We are working on the interface design with UWE right now to establish the best way to visualize our data output. It will probably include tables and maybe maps

Question 3: How do you see your service evolve on the long term (after the CURE project)?

We envision an application that bundles the air quality data with health data for a city, to be used as a decision tool for local governments, city planners etc. The application will need air pollution data from each city and can from there run the model to show health costs. The application should be made interactive and should have the ability for the user to tweak the different parameters such that e.g. the effect would be seen if all cars in the city went electric etc. and what that would mean in terms of monetary values of hospitalisations, mortality etc. and how diseases (cardiovascular and respiratory) are affected by different levels of air pollution. In the long term, once the research link between cause and effect becomes better known, the application should also include e.g. mental health, autism and diabetes. As well as the effects of air pollution on children aged 0-5, this is an important developmental stage in each person's life, and air pollution can have the ability to greatly negatively influence the brain and other bodily functions.

Question 4: What is the added value for scientists/urban stakeholders over existing services?

- Is there any commercial service available in the market that provides a similar solution? (i.e. which are the main "competitor services" offering alternative solutions to yours)? (Note: please list the "applications")
- What are their strengths and weaknesses comparing to your service?
- Do you know any new service in progress of development (with TRL 7 or 8) that could compete in the future with your service?

Denmark is in the unique position of having a central register with information regarding address, sex and age for all persons in the country. The CPR dataset (Central Personedata Register) is set into coordinates by matching the dataset with the national register of addresses, and making it into 1 km x 1km grid cells.

The EVA system describes the health effects of:

- Bronchitis in adults
- Bronchitis and asthma in children
- Sick days
- Hospital admissions for respiratory and cardiovascular diseases
- Lung cancer
- Acute deaths (caused by short term exposure)
- Years of Life Lost (caused by longterm exposure)
- Total amount of deaths (=chronic years lost/10,6* + acute deaths)

And includes NO_2, CO_2, $PM_{2,5}$ and PM_{10}, SO_2 and O_3.

There are two other known Air Pollution Health Risk Assessment Tools on a regional level, Aphekom and EcoSense. We refer to below tables for comparison of the three systems (Anenberg, S. C., Belova, A., Brandt, J., Fann, N., Greco, S., Guttikunda, S., … Van Dingenen, R. (2016). Survey of Ambient Air Pollution Health Risk Assessment Tools. Risk Analysis, 36(9), 1718–1736. doi:10.1111/risa.12540).

Table IV. Key Technical Characteristics of Tools with Regional (i.e., Multiple Countries in One World Region) Scope

Characteristic	Aphekom	EVA	EcoSense
Region:	Europe	Northern Hemisphere	Europe
National		x	x
City-level	x	x	x
Grid		x	x
Pollutants:			
$PM_{2.5}$	x	x	x
PM_{10}	x	x	x
Ozone	x	x	x
NO_2		x	x
SO_2		x	x
CO		x	
Other		Dioxins, mercury, black carbon	Heavy metals, dioxins, radio nuclides
Health outcome:			
Mortality (cases)	x	x	x
Disability-adjusted life years (DALY) or years of life lost (YLL)	x	x	x
Morbidity (cases)	x	x	x

Table V. $PM_{2.5}$ Concentration-Response, Population, and Baseline Incidence Data Sources for Tools with Regional Scope

Characteristic	Aphekom	EVA	EcoSense
$PM_{2.5}$ concentration-response relationship:			
User-defined		x	
American Cancer Society[56]	x	x	
Other	Several others[57–59]	WHO	WHO[47]
Population data source:			
User-defined	x	x	
Other		GEOSTAT	JRC population density grid[60] with enhancements
Baseline incidence data source:			
User-defined	x		
World Health Organization		x	
Other			EUROSTAT, national statistics

Table VI. Key Operational Characteristics of Tools with Regional Scope

Characteristic	Aphekom	EVA	EcoSense
Format:			
Software download		x	
Microsoft office program	x		
Web based			x
Open source	x		
Proprietary		x	x
Peer reviewed/policy applications:			
Peer reviewed	x	x	x
Used for policy applications	x	x	x

Other approaches:

Demographic data and life expectancy can be taken from Eurostat (population by age and sex, and life expectancy by age and sex), and the mortality data can be taken from WHO (European detailed mortality database). The exposure-response relationship and the population at risk can be selected by following a recommendation from the Health Risks of Air Pollution in Europe (HRAPIE project - WHO, Health risks of air pollution in Europe – HRAPIE project: New emerging risks to health from air pollution — Results from the survey of experts , World Health Organization). Quantifications of health impacts are done individually for these air pollutants, and they cannot be added together, as they exhibit some degree of correlation, positive or negative. For example, when adding together the values for PM and NO, this may lead to double counting of the effects of NO up to 30 %. These quantifications can then be directly converted to population exposure and premature deaths or years of life lost, but this does not provide a quantification of the health and economic impact (http://www.euro.who.int/__data/assets/pdf_file/0006/238956/Health_risks_air_pollution_HRAPIE_project.pdf?ua=1)

There is also a recent study funded by Public Health England (PHE) which has used a microsimulation model to produce longitudinal projections for the UK, but is not city specific and only looks at PM2.5 and NO2, hence lacking other crucial pollutants. This study used a dynamic microsimulation model to predict the future health and economic impact of the air pollutants $PM_{2.5}$ and NO_2 to the year 2035. This microsimulation examined data on air pollution exposure by age and sex, making use of disease and population data collected from the literature and from publicly available databases. The microsimulation method is an advanced method for modelling chronic diseases because of its capacity to simulate entire populations at an individual level over a lifetime.

The microsimulation model can be expanded to include additional pollutants and different populations but further work would be needed to develop the model. (https://journals.plos.org/plosmedicine/article?id=10.1371/journal.pmed.1002602)

Question 5: How easily can your service be upscaled to other cities (both in Europe and outside Europe)?

The EVA model can run on "default settings" for all cities in Europe, so in principle the upscaling could be fairly easily done. However, if a more detailed analyses is desired, local data is needed, such as the air quality data and the analyses can also be tailored better if there are local data sources on the inhabitants of the city (number of inhabitants + age + sex + living location).

Question 6: Is your service linked with existing Copernicus services?

Yes and No. Aarhus University and the EVA model is right now part of two Copernicus service projects, one of them is the CAMS52, however the model does not build on the existing Copernicus services but rather applies the model to these services as a prognosis for air quality.