PUBLICATION OF STRATEGIC NOISE MAPS IN ON-LINE NOISE MAPS PLATFORMS

Antonio Notario

DataKustik GmbH – Gewerbering 5, 86926 Greifenberg, Germany
email: antonio.notario@datakustik.com

According to the EU-Noise Policy, the Member States have to ensure that Strategic Noise Maps are made available to the public in a clear, comprehensible and accessible way. On the other hand the information should be updated every 5 years. Within such noise mapping projects, any modification of data which needs manpower can raise the time and cost enormously. A technique based on scripting is presented for automating time consuming tasks such as iterating over different configurations of calculations, summing up source-specific noise maps including level corrections or exporting maps to on-line Noise Map Interfaces.

1. Introduction

The main function of nose prediction software is the calculation of situations oriented to predict noise in open environments. These calculations can be applied to new future scenarios including the planned noise reduction measures such as noise barriers or new traffic distributions, to name few. On the other hand, the fast comparison of these new scenarios versus the situation before any noise reduction measure becomes a reality is a valuable feature that ensures the selection of the most efficient solutions in terms of noise reduction and total investment.

Calculations are executed by using national or international noise calculation standards; examples are the Interim Methods recommended by the EU which will be substituted by the new CNOSSOS-EU [1] Methods very soon. Either way, the strategic noise assessment in agglomerations includes a number of aspects to consider beyond the calculation techniques. As an example, strategic noise maps must be updated every 5 years in order to include new input data – e.g. new emission data such as traffics – as well as the noise reduction measures that have been implemented as a result of the approved Noise Action Plans during the previous stage. Another important aspect is that according to article 9 of the EC Directive [2] the Member States have to ensure that strategic noise maps and action plans are made available and disseminated to the public. In addition, the EC Directive demands: "This information shall be clear, comprehensible and accessible." In order to achieve the previous statement Member States take different solutions. Many of them choose the creation of an official platform [3] where results are exported in a GIS format and presented graphically. The main advantages are the homogenization of input and output data formats as well as the standardization of the procedure itself although implementation costs are usually higher provided outsourcing IT services are usually needed for the creation and maintenance – including the update of the noise map results – of the platform. This paper presents and discusses a project for the creation of an online noise map platform without previous web programming knowledge, which facilitates its startup and maintenance.

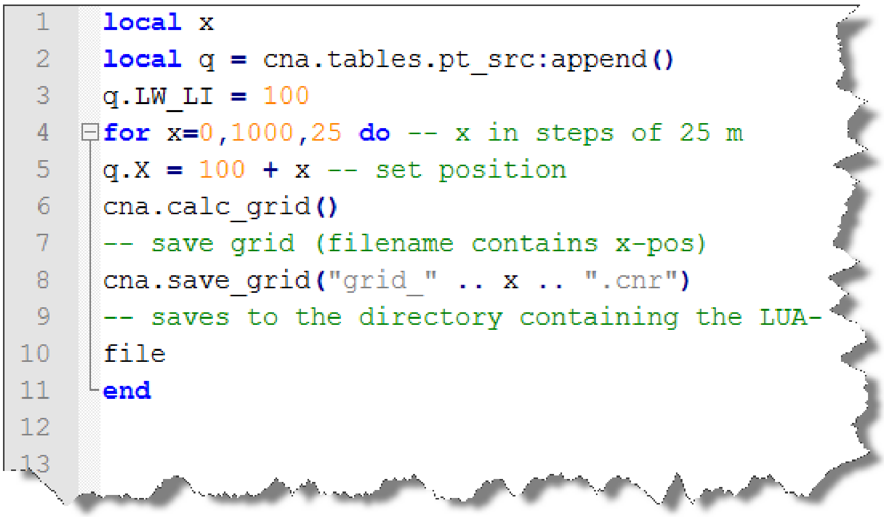
2. Basic concepts of Lua - Scripting

Lua [4] is a well-known and highly regarded extension language designed to be used by non-programmers and programmers alike. Its syntax is a reminiscent to *Pascal* and therefore familiar to many and easy to master. In the scope of using *Lua* for scripting commercial noise calculation software, a Lua interpreter has been implemented in CadnaA [5]. The interaction between the script and the calculation software is mediated by the global *cna* object. This object is only accessible when the script file is run from within the software. The *Lua* interpreter allows:

- The access to the attributes of the objects (through identifiers).
- The access to test variables.
- The Access to polygon vertices of any object.
- The execution of routines to perform operations within object groups.

Data types include comments, numbers, logical values used for comparison, character strings and Lua objects. It also includes various types of arithmetic, assignation, concatenation, comparison and logical operators. Moreover it allows the control of workflows and routines by means of a command group *if then else if else*. Finally functions and tasks can be defined directly.

As an example a Lua script is presented in Fig. 1. The following script adds a point source *q* to the project and moves it to a given position to calculate the grid. This process is repeated 400 times and with every step, the point source is moved 25 m on the x axis, therefore the resulting 400 numbered grid files show how sound levels change while the point source has been moved over the distance of 1 km.



```lua
local x
local q = cna.tables.pt_src:append()
q.LW_LI = 100
for x=0,1000,25 do -- x in steps of 25 m
q.X = 100 + x -- set position
cna.calc_grid()
-- save grid (filename contains x-pos)
cna.save_grid("grid_" .. x .. ".cnr")
-- saves to the directory containing the LUA-
file
end


```

Figure 1: Lua Script Example (comments are shown in green color)

After declaring the variables containing the x-position ***x*** and the point source ***q*** the counter is increased in steps of 25 from 0 to 1000. New objects can be created by appending the CadnaA table with the method ***cna.tables.*:append()*** – therefore ***q*** is a reference to a new object -. For each iteration step, the x attribute of the point source is modified and the grid is calculated and saved. In order to generate one grid of results per iteration, the expression ***grid_" .. x .. ".cnr*** *is used.*

Finally, resulting grids are saved to a specific directory, thanks to the command ***cna.save_grid*** followed by the directory where files are to be saved.

3. Lua Scripts and their application in noise mapping

As mentioned before, the application of *Lua* scripting allows the automation of repetitive and tedious tasks. Some real examples are presented in this section.

3.1 Simplification of Digital Terrain Models

In the context of big simulation projects where the original elevation model is too detailed it is necessary the application of techniques which allow the simplification of the data while the error limits are acceptable. The study carried out by DataKustik GmbH in the territory of North Rhine-Westphalia with the objective to identify areas where the construction of wind turbines is generally possible [6] had a size of 36,540 km² and the elevation model was delivered with a heavy detailed 10m x 10m grid. Thanks to the application of a *Lua* script the model was simplified in such a way that the final height deviation stayed within a predefined level. The whole simplification process was processed with a height-accuracy of 2 m and resulted in 2.6 million height-points after the simplification.

3.2 Assessment of level differences of different calculation standards

It is known, that with some noise calculation procedures, the sound pressure level at two receivers being close to each may differ quite substantially. With these procedures, the local level has a lower spatial accuracy than with other procedures. In order to evaluate this kind of scattering of results, the sound pressure level is calculated for small changes of a parameter (e.g. using the source coordinate x) using a *Lua* script. To this end, the noise level of N receivers that are all within a 1 m range around the receiver‘s original location shall be calculated. For those N noise levels, the arithmetic mean μ and the standard deviation σ are calculated and used to quantify the noise level distribution by applying the following equations:

$$\mu = \sum_{i=0}^{N} x_i \tag{1}$$

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=0}^{N}\left(x_i - \mu\right)^2} \tag{2}$$

This application could be used to assess the level differences when applying different configuration settings or calculation standards, for example.

3.3 Data conditioning

It is well known that the data conditioning easily represents a minimum of the 60% of the overall project´s timeline. This amount of time can be higher as it depends on the type and quality of the input of the imported data. In case no official data formats are available – e.g. GIS databases – or those formats are incomplete, inconsistencies such as building without heights can be found after the import. Even the different noise prediction software available usually implement tools for solving these problems, the situation could get more complicated as the project size becomes bigger. The automation of tasks can be of enormous help in those cases. Using the example of a project with a big number of buildings, a *Lua* script can be programmed to iterate over the building table in order to find those with a height value less than 0.1m and replace it with any valid height. This can be applied for any attribute and any object. Examples are the assignation of areas of land use or the deactivation of the subway network within a city.

4. Lua scripting applied to the creation of online noise maps

As an application example the use of a Lua Script in order to export a noise mapping project to an online and interactive noise map (ONM) is explained. The ONM platform consists of two parts: the platform code – basically a HTML/JS/CSS website – acting as a template and the data of the noise scenario, which is exported from the project files. The platform code is executed within an end-user browser and therefore, a plain and simple HTML server is enough for hosting the ONM platform.

4.1 Basics of the ONM platform

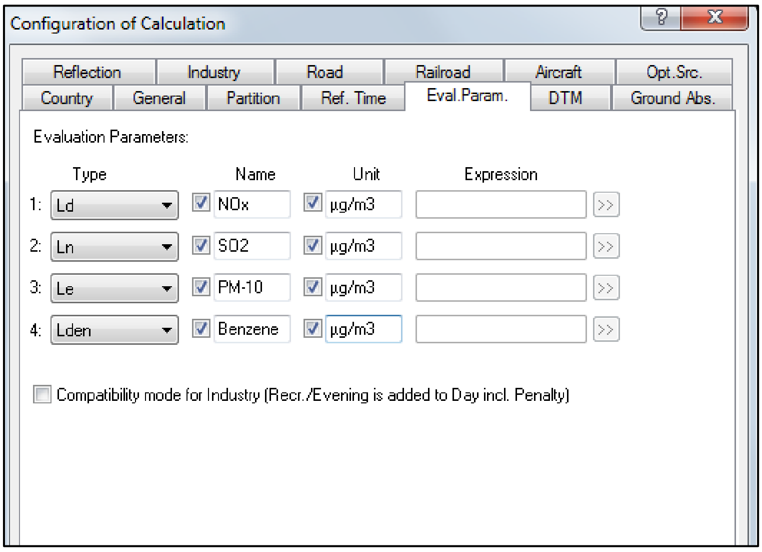
The ONM platform mimics the look of the noise prediction software platform in some way although there are important differences. For example, the ONM platform replicates the noise prediction software interface allowing the selection of variants and evaluation parameters but also allows the combination of noise evaluation parameters and air pollution results for each variant. Another difference is the navigation through the project. The ONM interface presents a navigation panel with predefined zoom levels each one with a different resolution. It can also include additional menus showing alternative information such as road network or city planning information.

4.2 Exporting ONM data from the noise prediction software

4.2.1 Preparation of the Models

The discussed example is a noise project called *project.cna* and consisting of four different scenarios with four evaluation parameters each. The last scenario contains the air pollution evaluation parameters NO_x, SO_2, PM-10 and Benzene which have been calculated in a grid with a different step size than the noise grids. First all those variants need to be calculated and then saved as independent project files: *projectv1.cna*, *projectv2.cna*, *projectv3.cna* and *projectv4.cna*.

The configuration dialog should be used to define Identifiers and units for the air pollution evaluation parameters because these will be the ones that are exported to the ONM data.



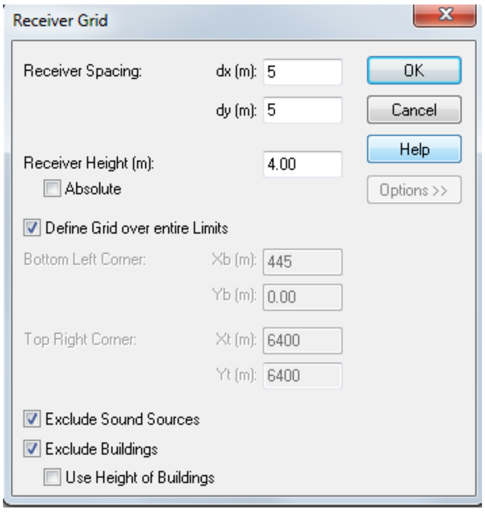
Evaluation Parameters:

	Type		Name		Unit	Expression
1:	Ld	☑	NOx	☑	µg/m3	
2:	Ln	☑	SO2	☑	µg/m3	
3:	Le	☑	PM-10	☑	µg/m3	
4:	Lden	☑	Benzene	☑	µg/m3	

☐ Compatibility mode for Industry (Recr./Evening is added to Day incl. Penalty)

Figure 2: Configuration of the Air Pollution evaluation parameters. The “name” field is used to indentify the parameters in the ONM platform.

Since the ONM platform is optimized for the web and its tiling technique approach is based on the grid limits and the distance between grid points, it is necessary to check both grid limits and grid point distances for all the variants. Due to the fact that usually noise grids have a different step size than air pollution grids there are discrepancies between the calculation points of both. If the mismatch is known a further grid oversampling will be needed.



Receiver Grid

Receiver Spacing: dx (m): 5
dy (m): 5

Receiver Height (m): 4.00
□ Absolute

☑ Define Grid over entire Limits

Bottom Left Corner: Xb (m): 445
Yb (m): 0.00

Top Right Corner: Xt (m): 6400
Yt (m): 6400

☑ Exclude Sound Sources
☑ Exclude Buildings
□ Use Height of Buildings

Figure 3: Receiver grid properties. Grid limits must match for all the project variants.

4.2.2 Preparation of the Lua Script

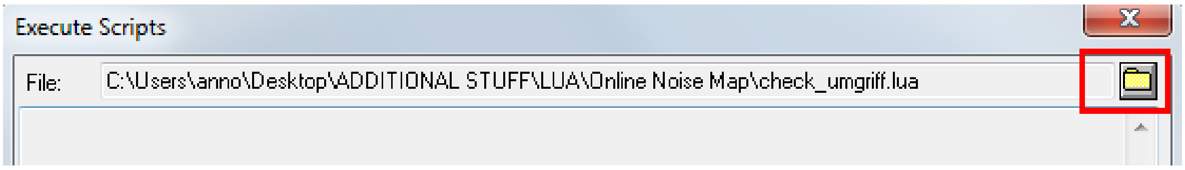
The export of data is performed by a Lua script. The objective here is the automation of the same operation on multiple files otherwise the operator would need to edit the original files several times and export manually each time. The main advantage of Lua scripts is that they can be adjusted to the individual needs of any project. As explained before in this project example several variants have been calculated for noise evaluation parameters with a grid spacing of 10m x 10m and the air pollution map has been calculated by using a 5 m x 5 m grid spacing. The script allows the desired oversampling so all grids are adjusted to the right spacing.

The structure of the script is divided into sections which are briefly explained below:

- Specification of the input – where the real Project is - and output– where results should be exported - paths.
- Definition of the variants. Within this section, the name of every variant that should be exported to the ONM platform is written. Each variant corresponds to one project file, including its corresponding grid. For example, the *projectv1.cna* is named as Scenario A.
- Check of the grid limits. The variants are previously loaded and a further check of the grid spacing is performed by writing out the limit coordinates.
- Exporting Routine. Finally variants are exported to Web-Bitmap format. This specific format creates a set of image tiles in different resolution steps which are coincident to the number of zoom steps that the ONM interface will have available. In this section an automatic readjustment of the grid spacing will be executed in case of grids with different spacing.

4.2.3 Running the Lua Script

Lua scripts are loaded and executed directly from the noise prediction software by using the corresponding direct command. There is no need to install any external *Lua* environment since the interpreter is built into the software:



Execute Scripts

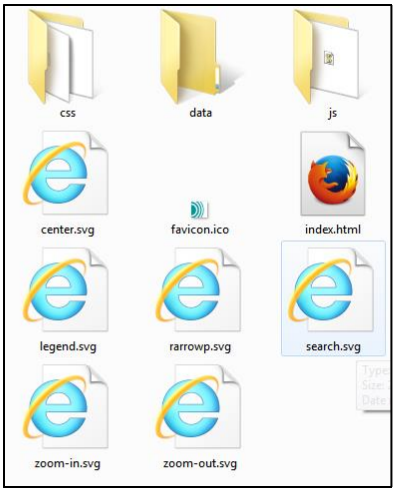
File: C:\Users\anno\Desktop\ADDITIONAL STUFF\LUA\Online Noise Map\check_umgriff.lua

Figure 4: Assignation and execution of the Lua Script

As a result of the export many information – map tiles exported as bitmap tiles with different resolutions - will be saved into the output path.

4.3 Data integration into the ONM platform template

The last step is the integration of data into the ONM platform. Since it is a template, the platform always has the folder structure shown in Fig. 5:



css

data

js

center.svg

favicon.ico

index.html

legend.svg

rarrowp.svg

search.svg

zoom-in.svg

zoom-out.svg

Figure 5: Folder structure of the ONM platform. Exported files need to be saved into the “data” folder

All exported files are simply and directly – without using sub folders - saved into the folder named “data”.

4.4 Description of the ONM platform interface

Starting up the ONM platform project only needs the allocation of the folder structure shown in Fig. 5 onto a web server. For testing purposes a local webserver on the computer is enough.

Once the ONM has been started, the web interface is accessible from any web browser and offers different features like the display of the color legend and the navigation panel. In addition there are drop down menus to switch over the different variants and evaluation parameters – including those related to air pollution -. In all cases the calculated value is updated and displayed in a window close to the mouse cursor.

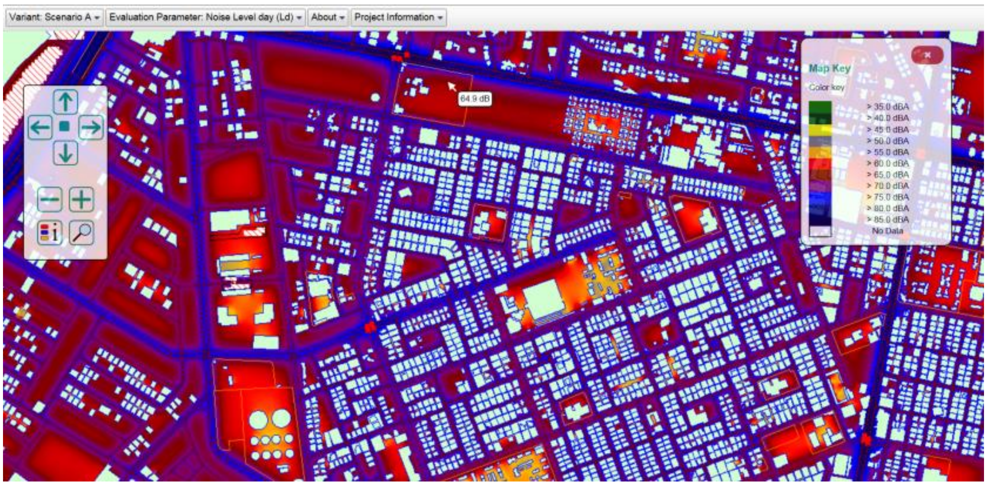


Figure 6: Online Noise Map web Interface. Noise Map L_{day} of Variant A

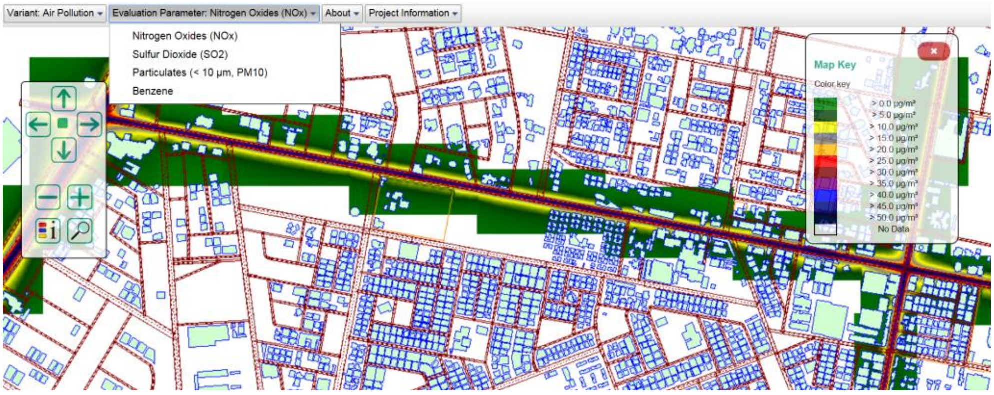
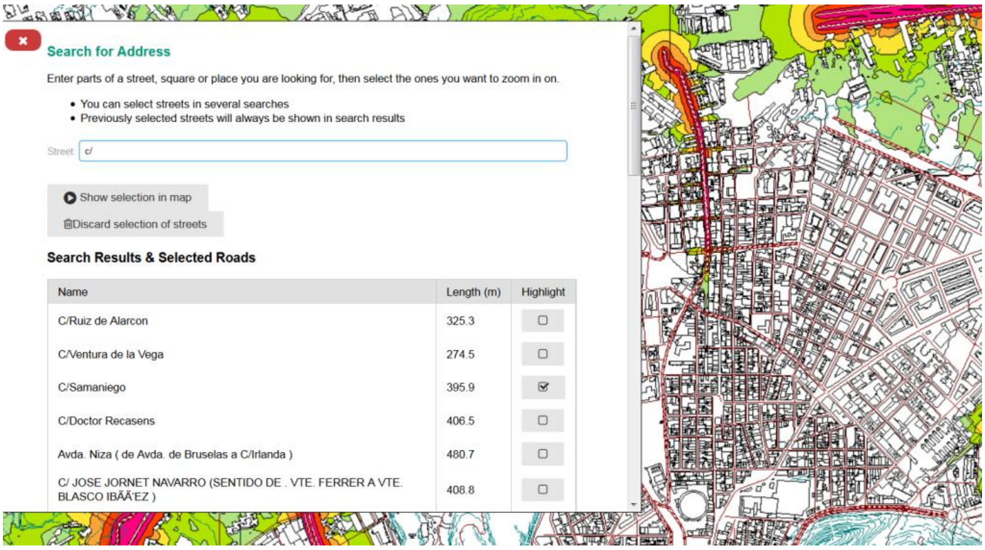


Figure 7: Air Pollution Map for NOx

The ONM web interface project also includes some features not present on the noise mapping software. The most relevant is the search address function to look for streets provided their name was included project file under the name field of the roads. This feature is in line with the requirements of the article 9 of the EU Noise Directive [2]:



Search for Address

Enter parts of a street, square or place you are looking for, then select the ones you want to zoom in on.

- You can select streets in several searches
- Previously selected streets will always be shown in search results

Street: c/

Show selection in map

Discard selection of streets

Search Results & Selected Roads

Name	Length (m)	Highlight
C/Ruiz de Alarcon	325.3	□
C/Ventura de la Vega	274.5	□
C/Samaniego	395.9	☑
C/Doctor Recasens	406.5	□
Avda. Niza (de Avda. de Bruselas a C/Irlanda)	480.7	□
C/ JOSE JORNET NAVARRO (SENTIDO DE . VTE. FERRER A VTE. BLASCO IBÃÄ'EZ)	408.8	□

Figure 8: Search for Address dialog. The ONM shows on screen the search result of streets present in the model.

5. Conclusions

The automation of noise prediction software tasks through scripts applied to noise prediction software demonstrates to be a powerful tool providing additional features usually not present in noise prediction software user interfaces. This project demonstrates a summary of benefits of these techniques applied to the presentation of noise mapping projects to the public compared to the standard procedures for creating online noise map platforms:

- Easy and fast implementation of noise mapping projects online.
- Small investment provided programming costs are negligible.
- In case the map must be updated as a result of a new calculation, the old exported files are replaced with the new ones. Therefore updating an online noise map is a simple copy and paste action.
- ONM platforms are easy to use by the public and they also include features which are usually not available in any noise prediction software as for example the “search for address” functionality.

REFERENCES

1 Commission Directive (EU) 2015/996, *establishing common noise assessment methods according to Directive 2002/49/EC of the European Parliament and of the Council*, (2015).

2 Directive 2002/49/EC of the European Parliament and of the Council, *relating to the assessment and management of environmental noise*, (2002).

3 Spanish Noise Pollution Information System (SICA), *Ministry of Agriculture and Environment*, (2015) [Online.] available: http://www.sicaweb.cedex.es

4 Information about Lua language [Online.] available: http://www.lua.org/docs.html

5 DataKustik GmbH, CadnaA – Software for Environmental Noise: www.datakustik.com

6 Probst F., Probst W. and Huber, B. Large-Scale Calculation of Possible Locations for Specific Wind Turbines under Consideration of Noise Limits, *Internoise* Innsbruck, Austria, 15–18 September, (2013).