

TIN Visualization (Triangulated Irregular Network) in Marushka Design



GEOVAP

OBSAH

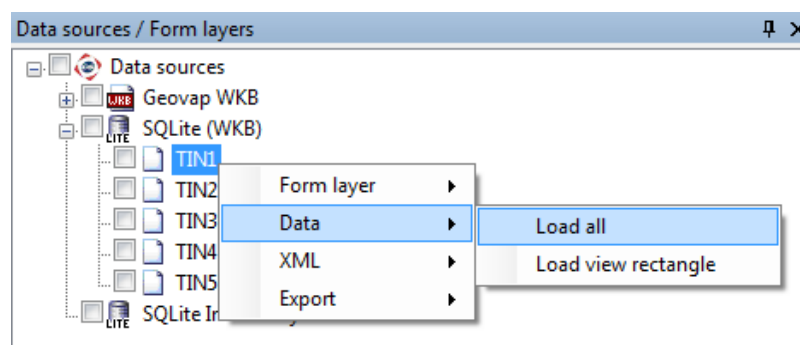
1	AIM OF THE EXAMPLE.....	2
2	WORKING WITH EXAMPLE	2
3	DIALOGUE BOX SAMPLE	3
4	A BRIEF DESCRIPTION OF THE EXAMPLE IN MARUSHKA DESIGN	5

1 Aim of the Example

The aim of this example is to demonstrate how to create and publish a triangulated irregular network (3D terrain model) from coordinates x, y, z. This example was created in version 4.1.0.3 and it does not have to be compatible with older versions.

2 Working with Example

- Unzip the **TIN_EN.zip** into **c:\MarushkaExamples** folder. The target folder must be respected due to interconnection of paths with the project. In the case of placing the files in the different folder, it would not be possible to work with example.
- Open the **TIN_EN.xml** in Marushka Design environment.
- Select form layer TIN1 in SQLite (WKB) data store, in the context menu choose Data – Load all:



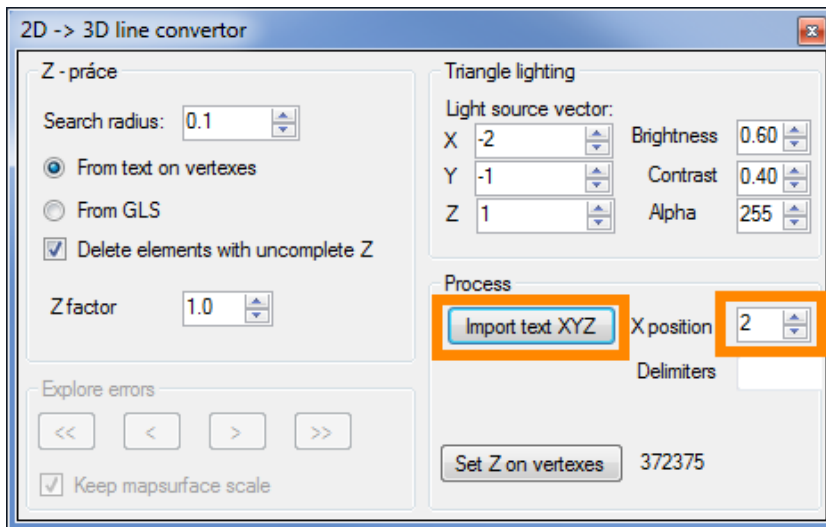
- In map window choose: "Fit all":



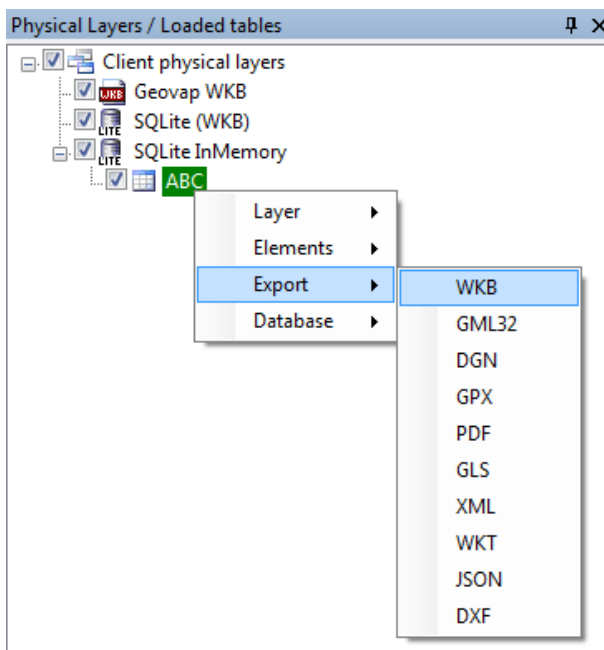
- Launch the local web server:



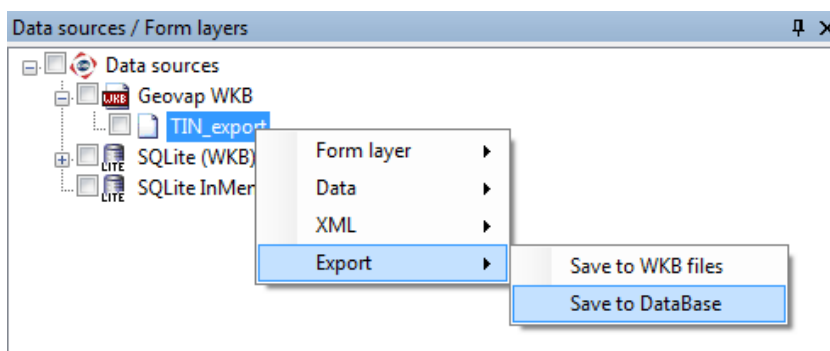
3 Dialogue Box Sample



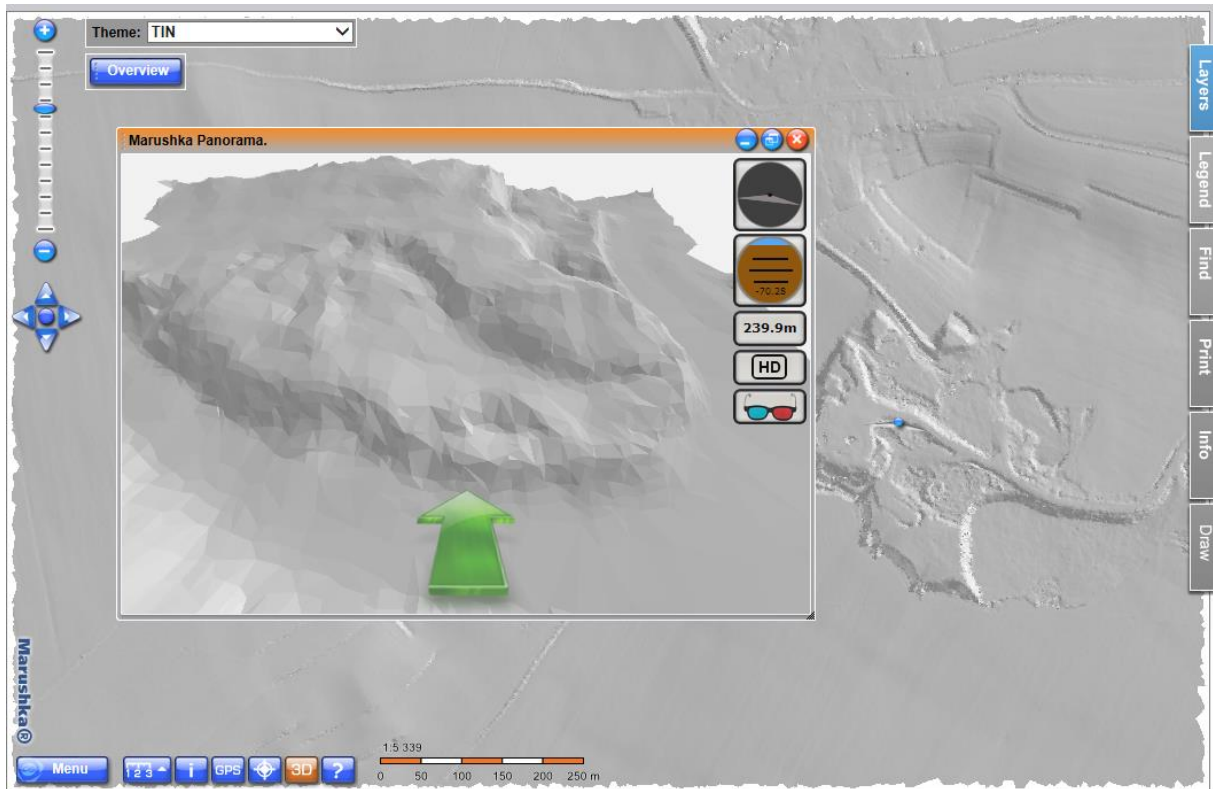
Img 1: Dialogue box 2D -> 3D line convertor



Img 2: Export to WKB



Img 3: Export from WKB to SQLite



Img 4: Map window sample with 3D panoramic view



Img 5: 3D panoramic view dialog box


4 A Brief Description of the Example in Marushka Design

This example contains a database in *SQLite*, into which is imported data from the source file. It also contains the *WKB* data store, which encompasses identical data as the *SQLite* database. This file is used only as an auxiliary format while exporting data, but thanks to its use is significantly shortened the time needed for export. The last data store in this example is *InMemory SQLite*, which is the data store that contains only the query, calling the 3D panoramic view. It is therefore only auxiliary data store.

Before starting work with this example, it is necessary to delete all the themes in your current Marushka Design installation. In the query that calls a panoramic view, is firmly set `themeid=11`. For this reason it is necessary to delete other themes.

Import of Coordinates in Marushka Design

To successfully import points into *SQLite* database, it is necessary to have installed module *WkbZ.WkbZ* (2D -> 3D convertor), which is an application for import of these coordinates. The dialogue box is thus started from the main menu *Modules – WkbZ.WkbZ*, you can see the dialogue box sample in *Img 1*.

It is necessary to set position of *X* coordinate (parameter *X position*). Specifically in this project, it is value 2, because the *X* coordinate is in the second position. Next, click on the button *import text XYZ*, but you need to have previously selected an arbitrary data store. It is also important to select the appropriate file (*CBRO21_5g.xyz*) and in Marushka Design click on the button *All* . This action will load all the points in the map window.

Export into WKB

In the next step, you need to export these points to a *WKB* file. This step still could be omitted and the points could be imported directly into the database, but without this step, the whole process would be a far lengthier. Now move to window *Physical layers / Loaded tables* in the data store, which was selected in the previous step, will appear the physical layer, in which are loaded points with coordinates. To export these data is needed to right click on this physical layer and select *Export – WKB*, see *Img 2*. Thus the elements are stored in *WKB*.

Export into SQLite database

Now the user will connect the *WKB* data store, which he previously created, into the project. Then perform a right click on the form layer containing the data in the *Data Sources / Form layers* window and select *Export – Save to Database*, see *Img 3*. Now it is necessary to select the form layer from the database data store. Subsequently, Marushka Design asks, if you want to keep *Ids* of elements. In this case it is not necessary, so click on option *no*.

After import of elements, whose progress can be monitored down on the status bar, it is necessary to connect this database source in the project.

Data Publishing

For publication of data is needed in the layer, which will display triangulated points, set in category 2. *Scale, order and symbology ~ Symbology*, subcategory *Triangulation*, parameter *Use* to value *True*. In category 1. *Identification and description(s)* it is necessary to set item *Dimension* to value *Space_3D*.

After these steps, it is possible in MarushkaDesign to load triangulated points (triangles) into Marushka Design map window. This action can be performed using operation *Data – Load view rectangle*, operation *Data – Load all* would load only points without triangulation into the map window. This measure is used to save time, because loading all the triangles in the project would take too much time.

Additional information

In the sample data store SQLite (WKB) are diluted individual points. Form layer TIN1 contains all the points, TIN2 contains 4* diluted points, which means that every fourth point is randomly displayed. Layer TIN3 displays every 8th point and TIN4 every 16th and TIN5 every 32nd point. This measure was implemented due to the fact that if every time should be loaded all the points, it would take a lot of time.

The source data are in layer TIN1. Tables TIN 2 – TIN5 are then filled with data using SQL: `Insert into TIN2 Select * from TIN1 where id % 4=0`. The red number 4 corresponds with the dilution value.

After starting the project, it is possible to start panoramic view using 3D button, controls are identical as in other Marushka 3D projects.