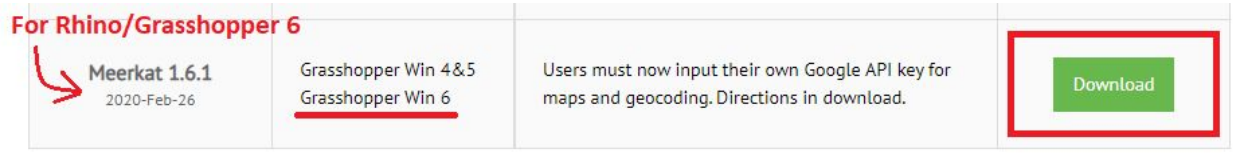
Site terrain using Meerkat for Grasshopper

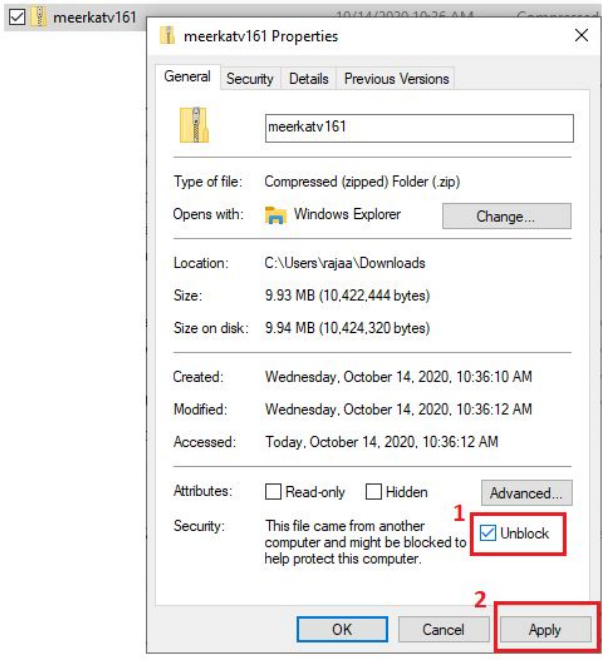
Meerkat is a free plugin for ***Grasshopper*** to process GIS data from shape files.
Meerkat is developed by ***Nathan Lowe*** and is currently under active development.
Note that the following workflow was documented in December, 2020.

Download and setup

Download ***Meerkat*** zip file from ***Food4Rhino***:
https://www.food4rhino.com/app/meerkat-gis



Make sure to ***Unlock*** the downloaded zip file before extracting the content by right-mouse-click, then go to ***Properties***. In the dialog, check the ***Unlock*** checkbox, then click ***Apply***:



If you do not have one, create an account with ***Google Cloud Platform***. You can set up a trial (90 days) and you will be given $300 credit to start with. You will need to provide an address and a credit card information to set up the account. You will not be charged automatically after using your credit and you will need to manually activate the automatic payment of $200 per

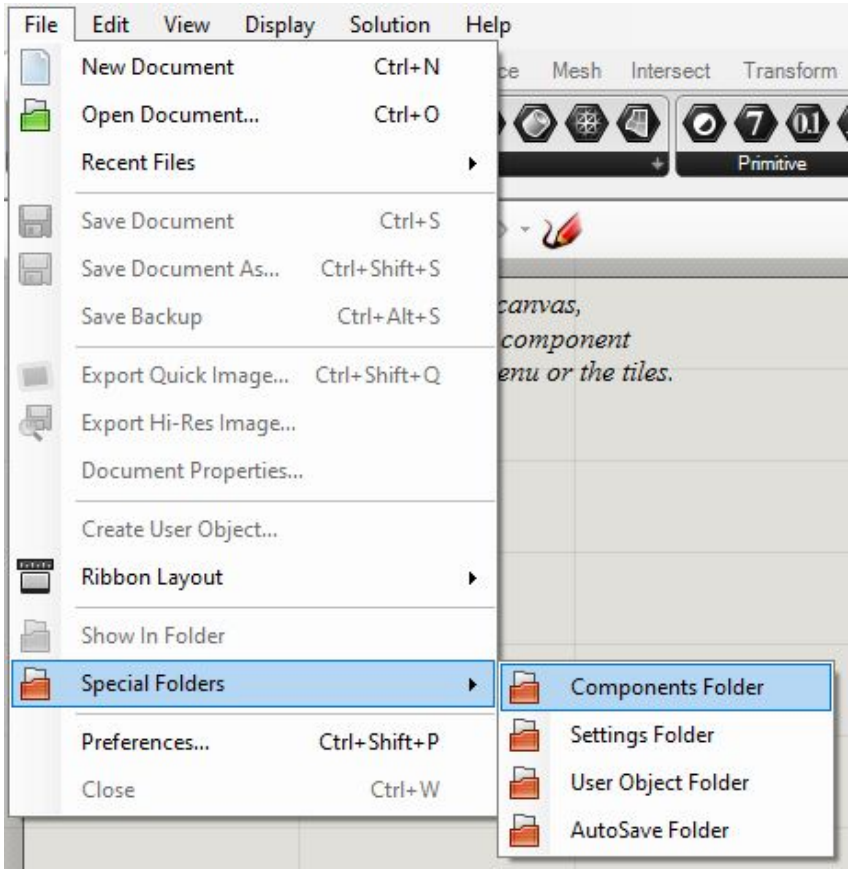
month for access.

Google Cloud Platform

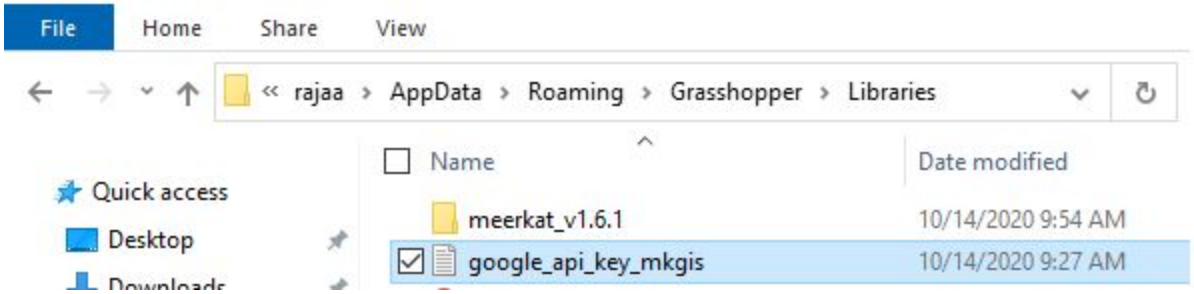
Locate your ***API key*** in your ***Google Cloud Platform*** . Make sure it is either not restricted to specific apps, or create an API key that has access to:

- ***Maps Javascript API***
  https://developers.google.com/maps/documentation/javascript/get-api-key
- ***Geocoding API***
  https://developers.google.com/maps/documentation/geocoding/get-api-key

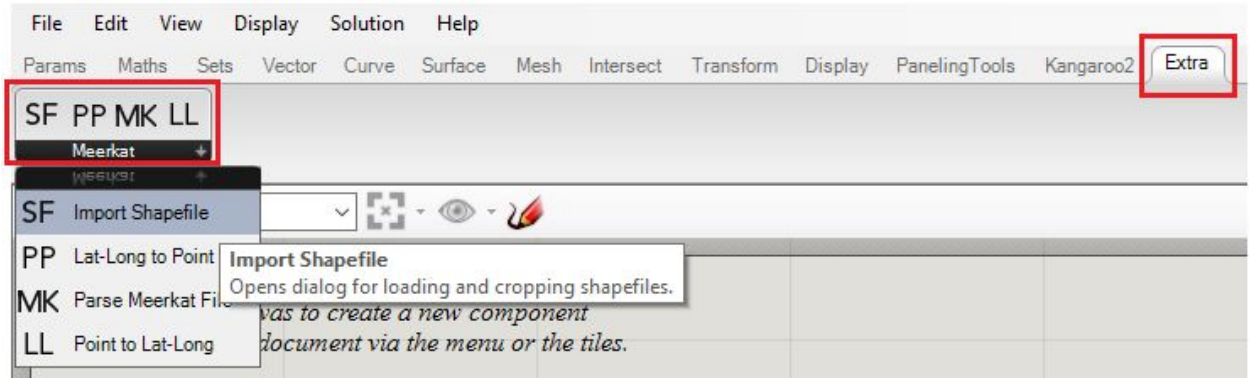
Copy your API key to the ***google_api_key_mkgis.txt*** file in the ***Meerkat*** folder (delete everything else in the file. You should only have the key there).
Move the file to the ***Grasshopper*** component libraries folder (you can open it through an open session in ***Grasshopper*** under ***File>Special Folders> Components Folder***.



Make sure you place ***google_api_key_mkgis.txt*** directly in the ***Libraries*** folder and not in a sub-folder, then move the extracted Meerkat folder and place it in the same Libraries folder.

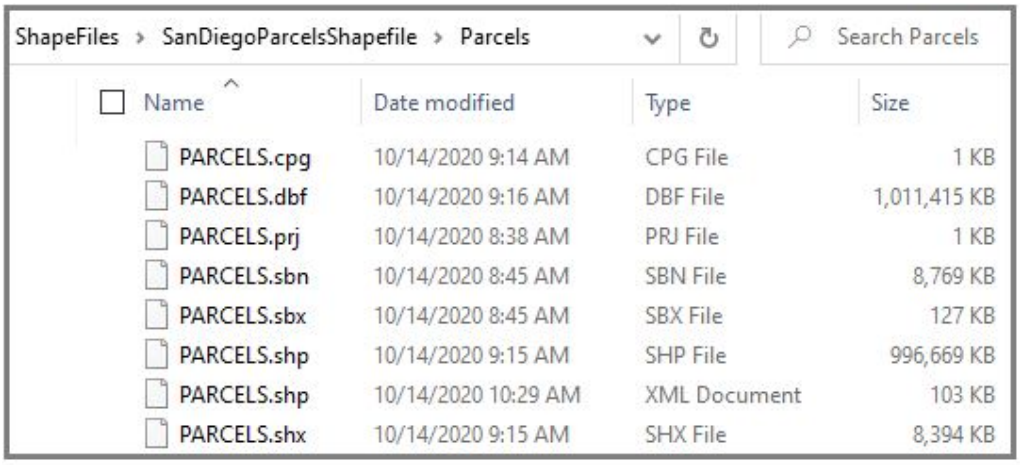


Open Grasshopper, and you will see a new ***Extra*** tab with ***Meerkat*** components inside:

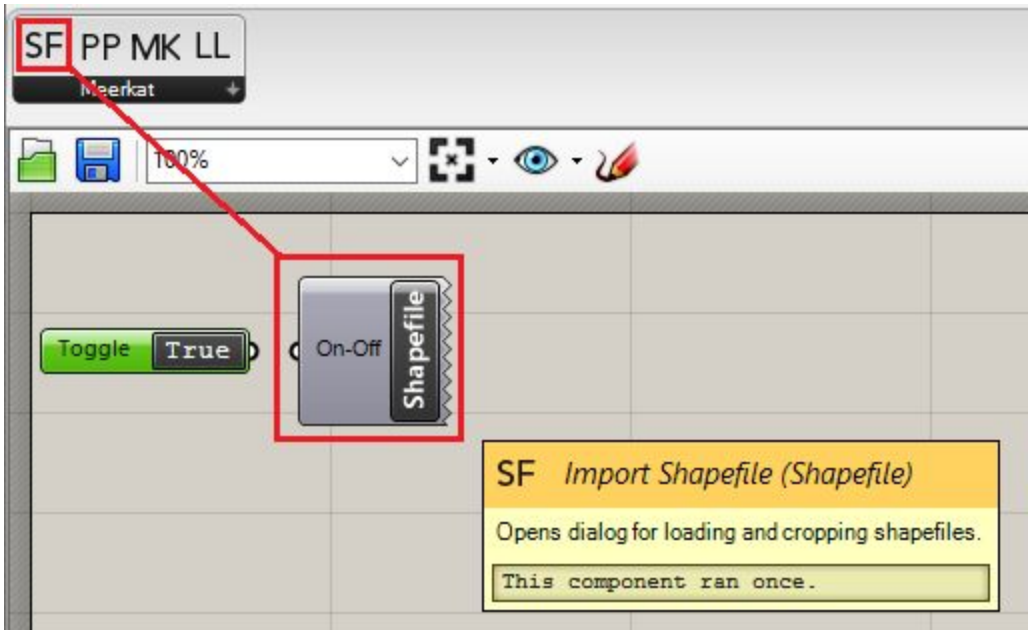


Meerkat Workflow

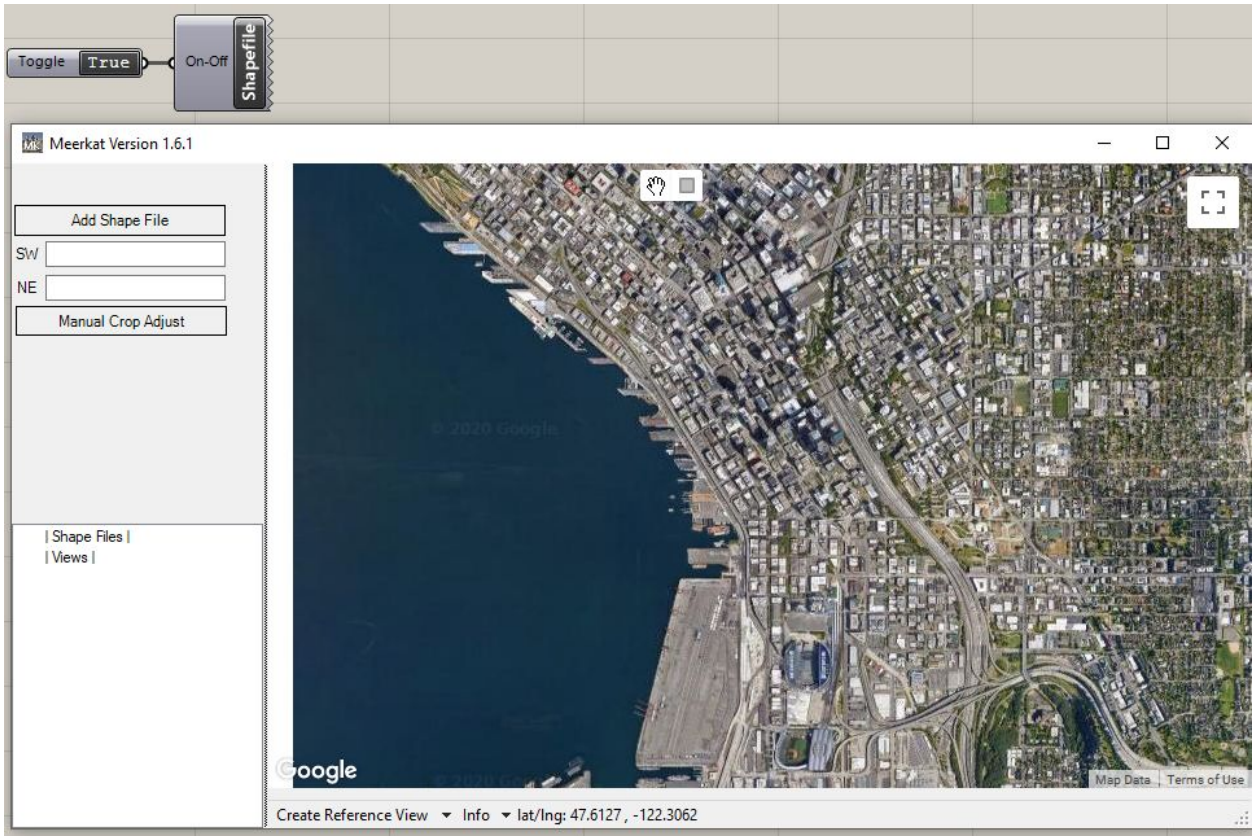
The first step is to download the shapefiles of the areas you are interested in. Different cities usually manage their shape files and each has a different location to access. For this example,we will get the ***Torrey Pines*** area in ***San Diego*** through ***sandag.org > Resources > Maps and GIS*** to get the parcels shapefile. Make sure to unlock the ***parcels.zip*** file before extracting data as explained above. The folder of the parcels typically include many files but only one ***.shp***.



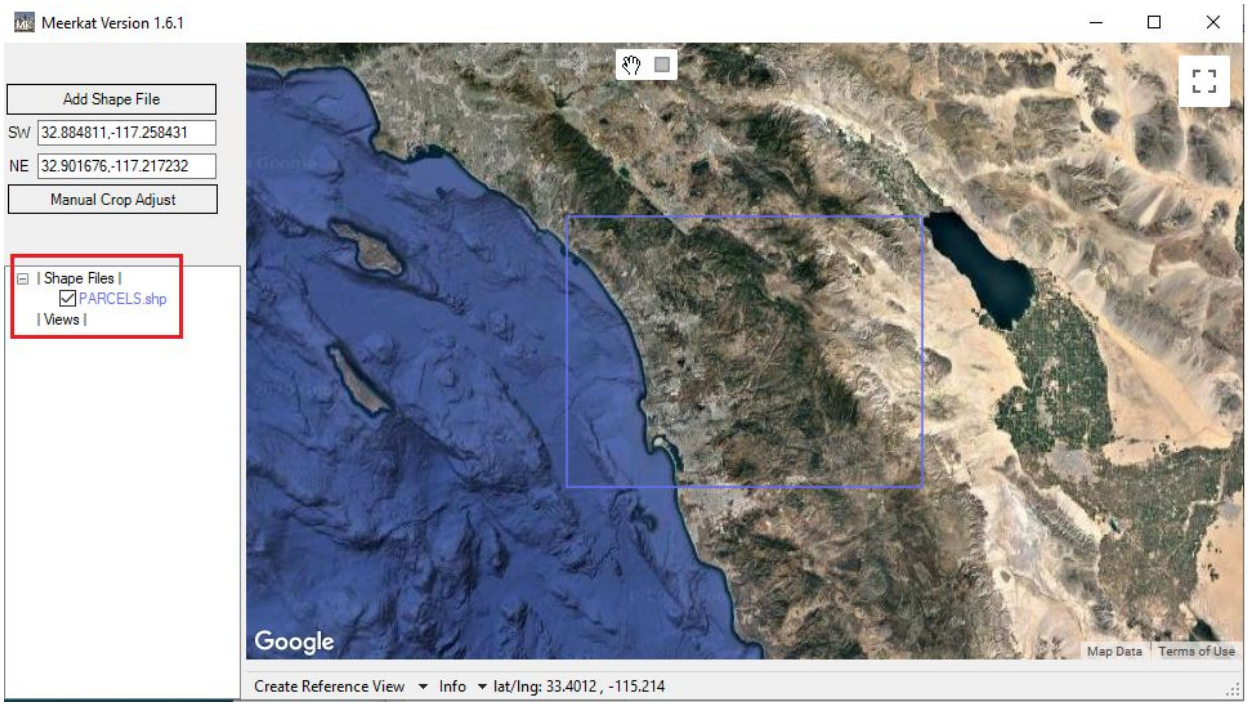
Using ***Meerkat*** plugin for ***Grasshopper***, you can use the import shapefile component to load your shapefile. Connect a GH ***Toggle*** to the ***Shapefile*** component



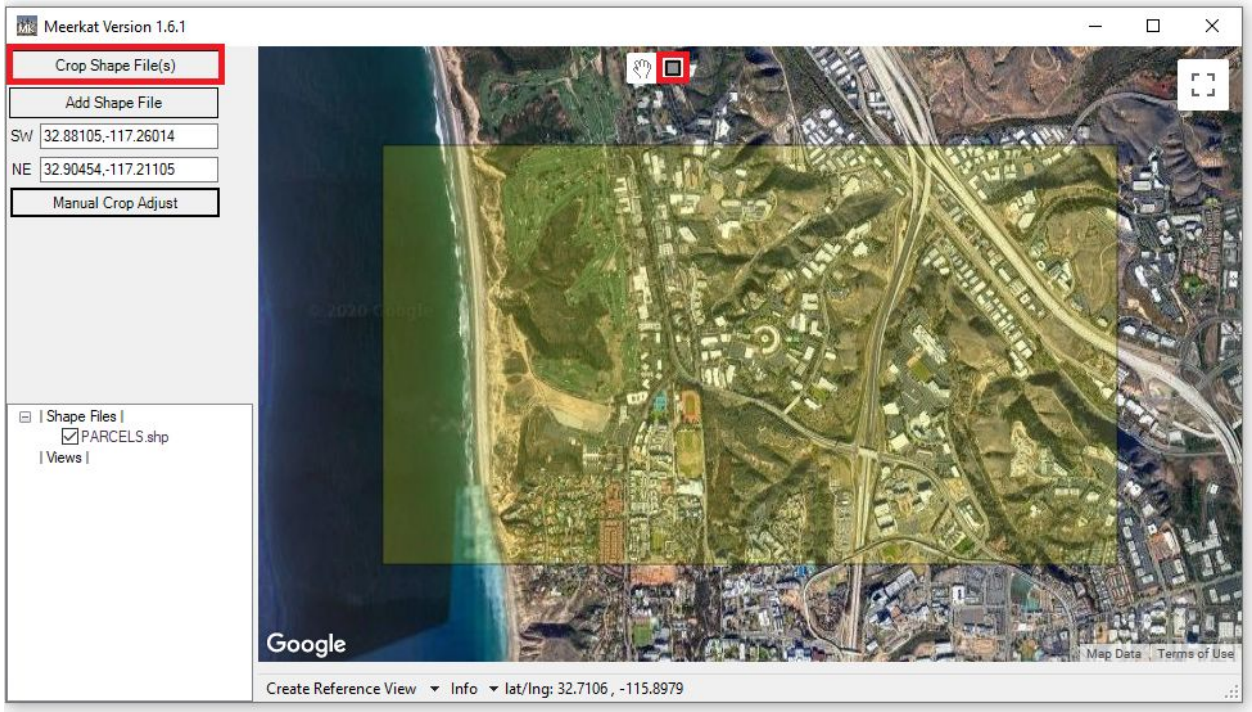
Once you connect the ***True*** toggle to ***Shapefile*** component, the following window opens. Clock on ***Add Shape File*** and navigate to your shapefile.



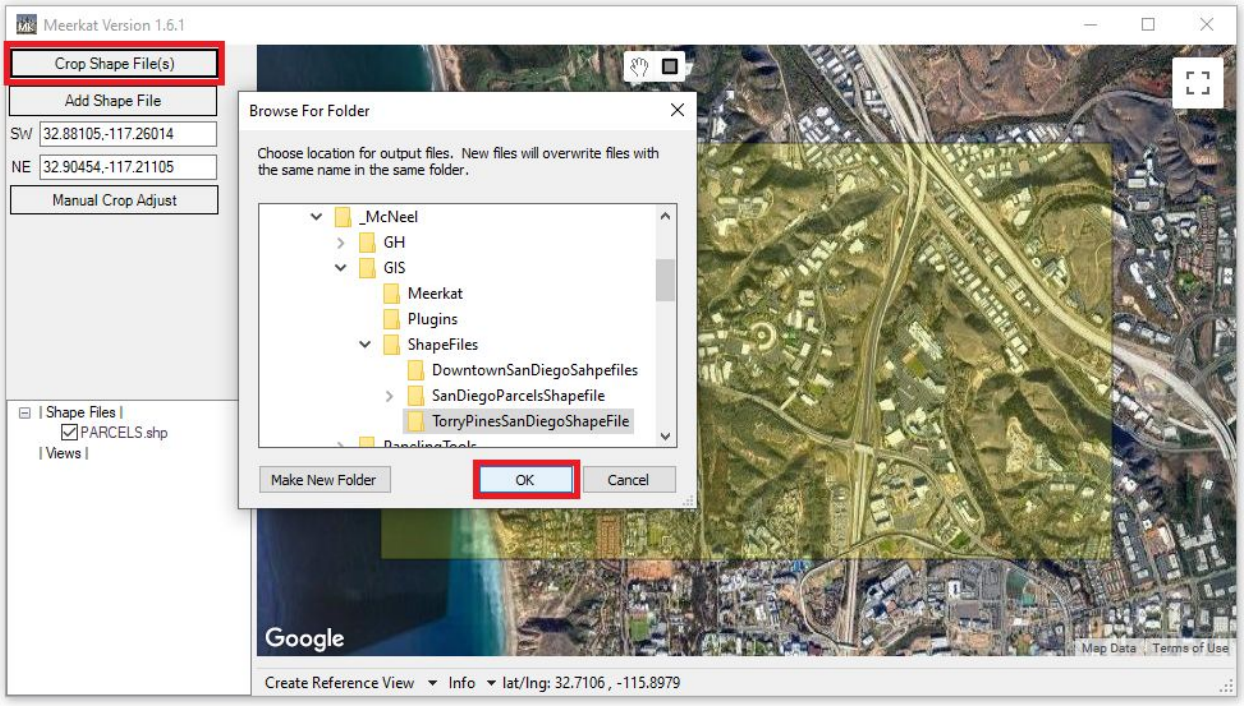
Once you add the shapefile downloaded earlier, you will see it added to the left menu, and when you check the box next to it, the map updates:



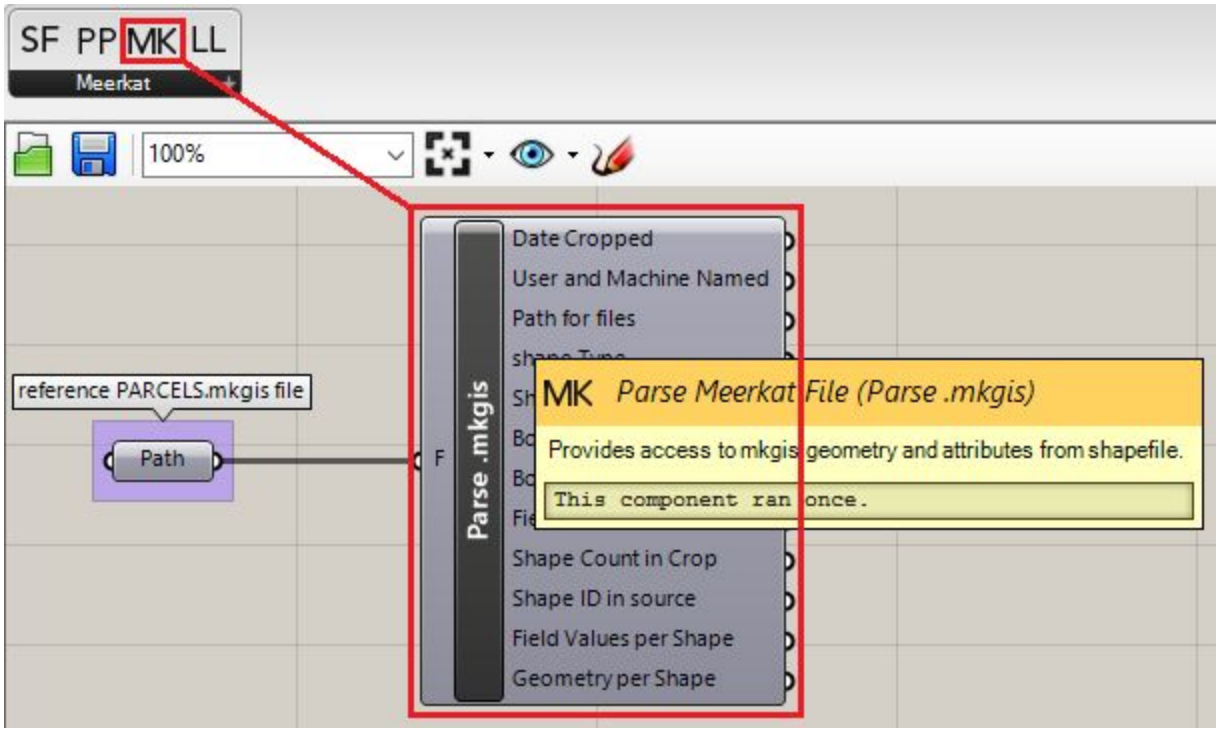
Zoom to the downtown area, then draw a boundary using the little square icon at the top of the map area. Notice that the left menu updates with the corresponding bounding coordinates. Click on the ***Crop Shape File(s)*** to select the marked area.



Create a folder for the cropped area to place your cropped shapefile. This will create a ***PARCELS.mkgis*** file in the selected location that you can then reference and use to extract data in ***Meerkat***.



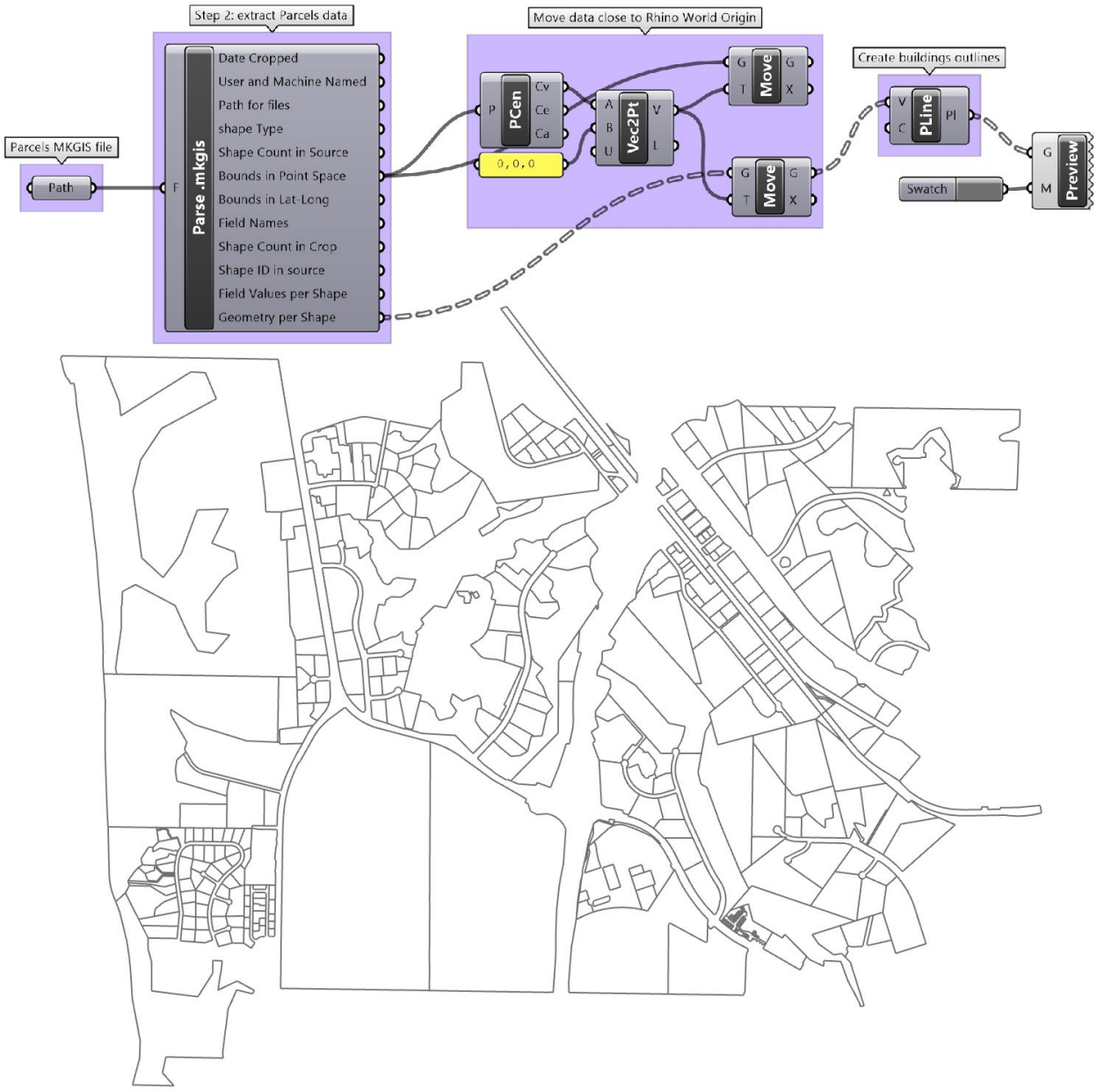
Now that the Meerkat shapefile is created, you can reference it to extract data. Use the ***Path*** component in ***Grasshopper*** to set the path and connect to the ***Parse .mkgis*** component



The information of the shapefile includes the boundaries of the parcels as an output from the ***Geometry per Shape*** in the ***Parse .mkgis*** component. In the Rhino viewport you can locate the points if you zoom to the right area. Most of the time, the data is placed away from the origin

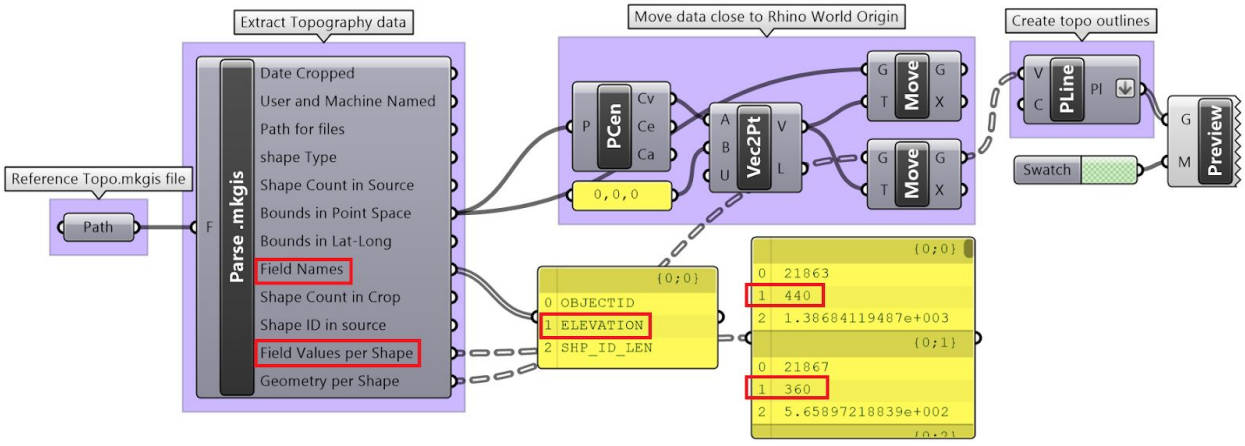
and this makes it hard to find and process in most cases. It is recommended to move all the points close to the origin before processing further.

The outlines of the parcels or sites are defined as point sets in the shapefile, therefore you can use the ***Polyline*** component to create the outlines of the buildings and terrain.

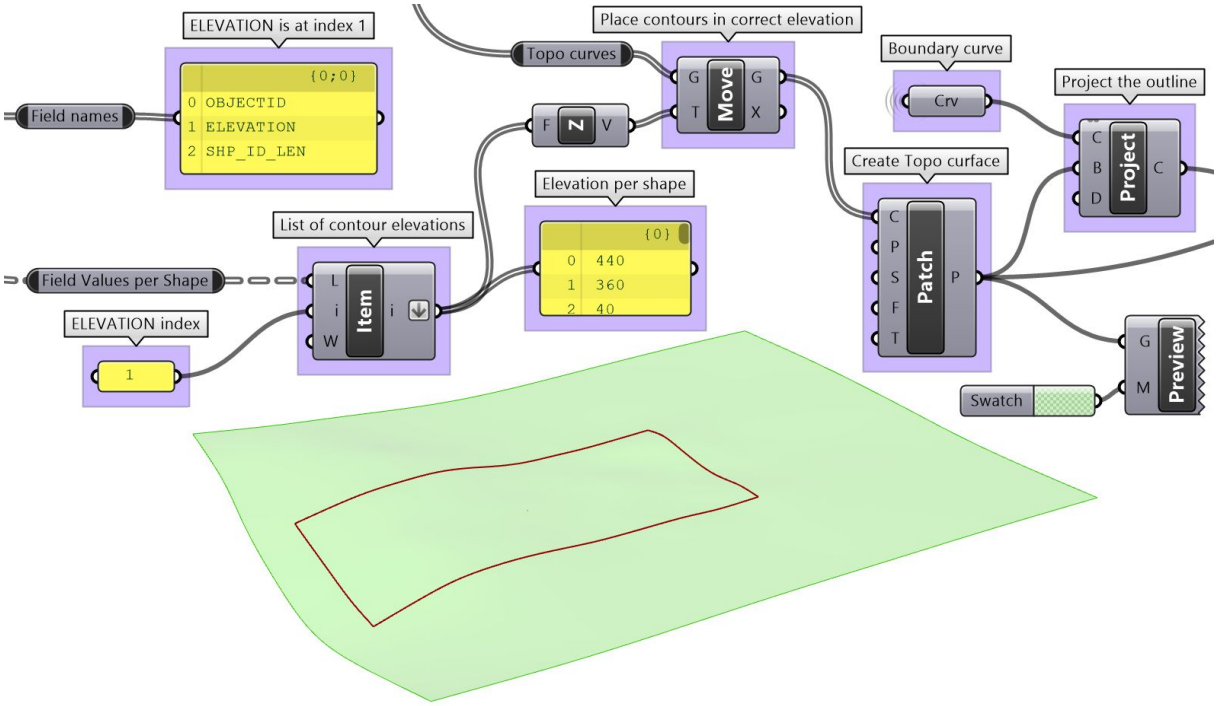


When adding and crop additional shape files that have other information such as topography, you need to make sure to apply the same boundary. You can create an approximate boundary then manually enter the exact location used earlier then use ***Manual Crop Adjust*** to create identical boundary, then click ***Crop Shape File(s)***. Alternatively add all shape files for the project and crop together.

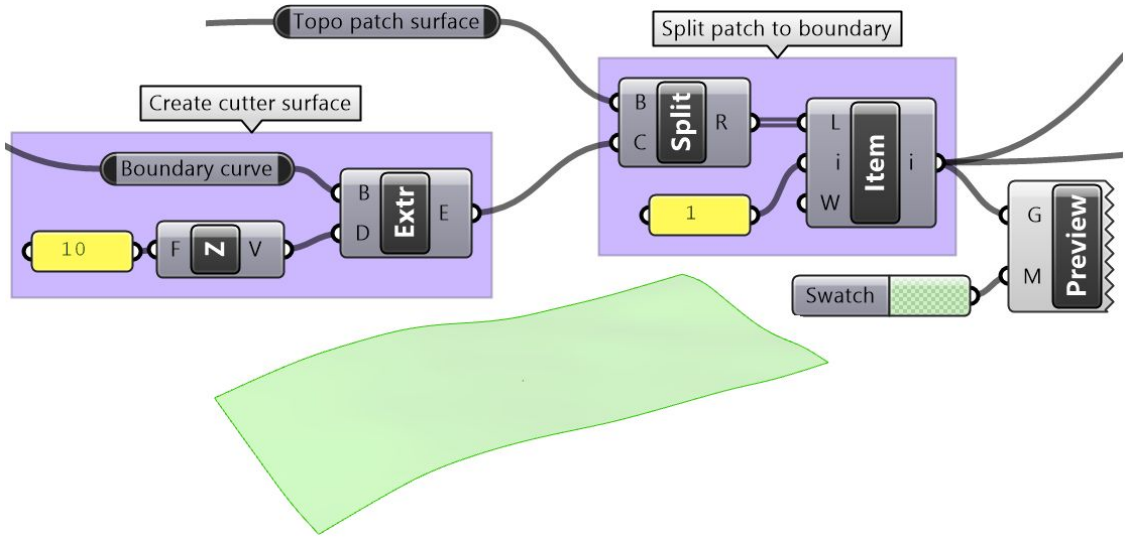
Once the topography is added to the Grasshopper file, you can move to origin to overlap properly with the parcels. Also notice that topography contours have elevation data. You can find the index of the elevation by examining the ***Field Names*** then extract the actual values from the ***Field Values per Shape***.



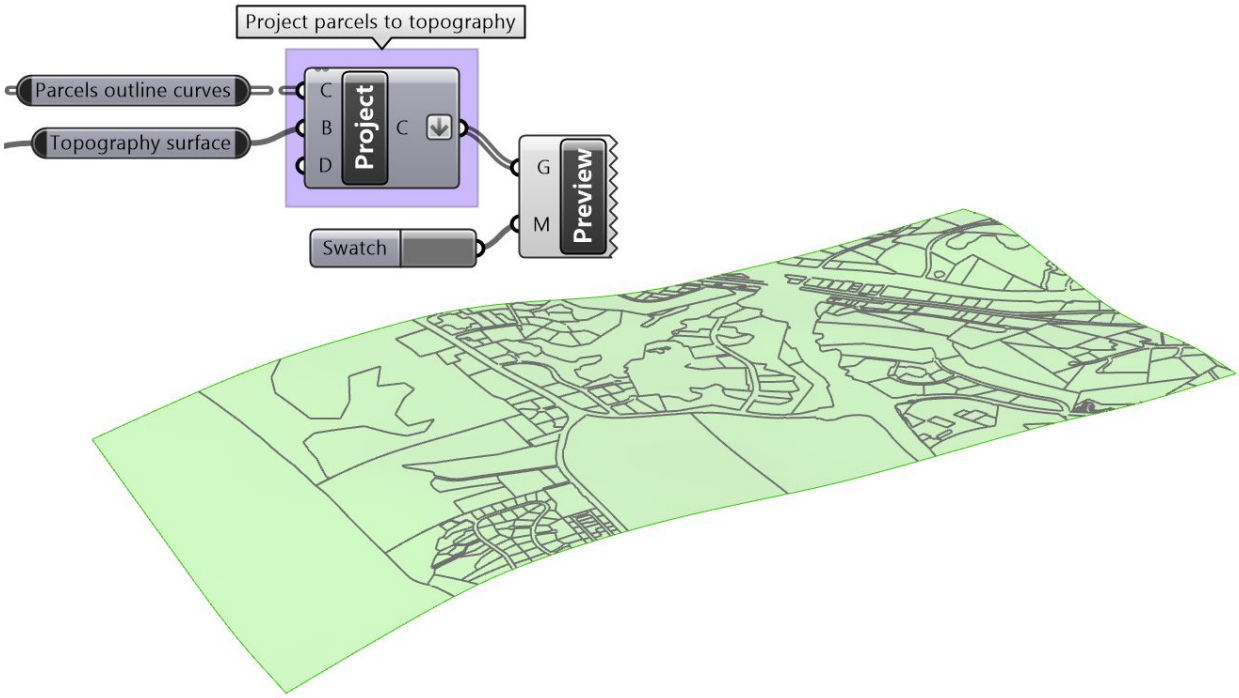
Use the elevation data per contour to place in the proper vertical location, then use ***Patch*** to create a surface through the contours. You can then project the outline to the patch.



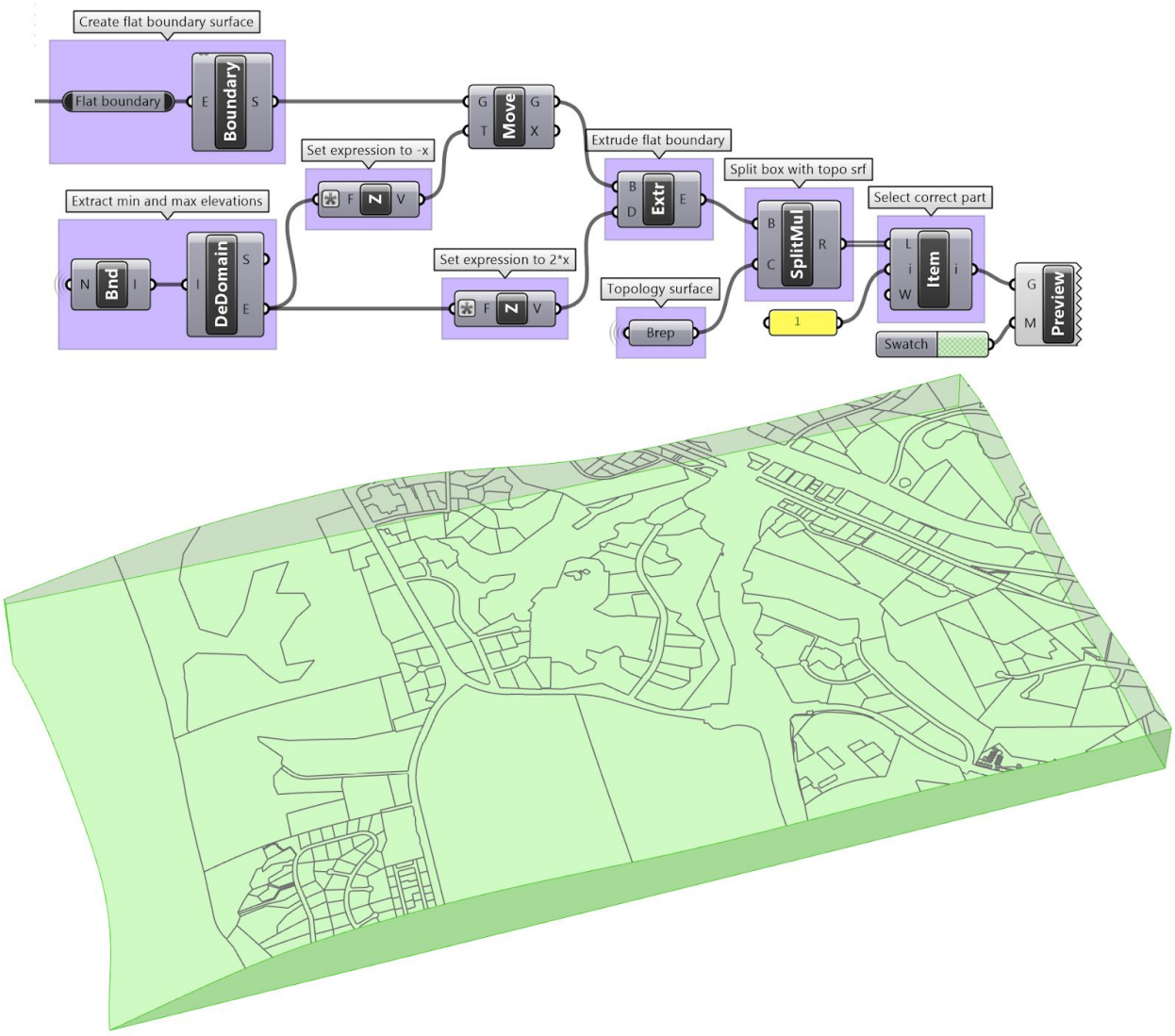
Extrude the outline and then crop the patch.



Next, project the parcels outlines onto the topography surface.



The next step is to generate the solid for the topography.



If you acquire the shapefile for the surrounding buildings, then you can also create the .mkgis and add to the model the same way as the parcels and topography, You can then create the 3D geometry of the surrounding buildings by extruding their footprint using the elevation value.