aerofly FS 2: Rodeo's Tutorial
Adding Custom Airports

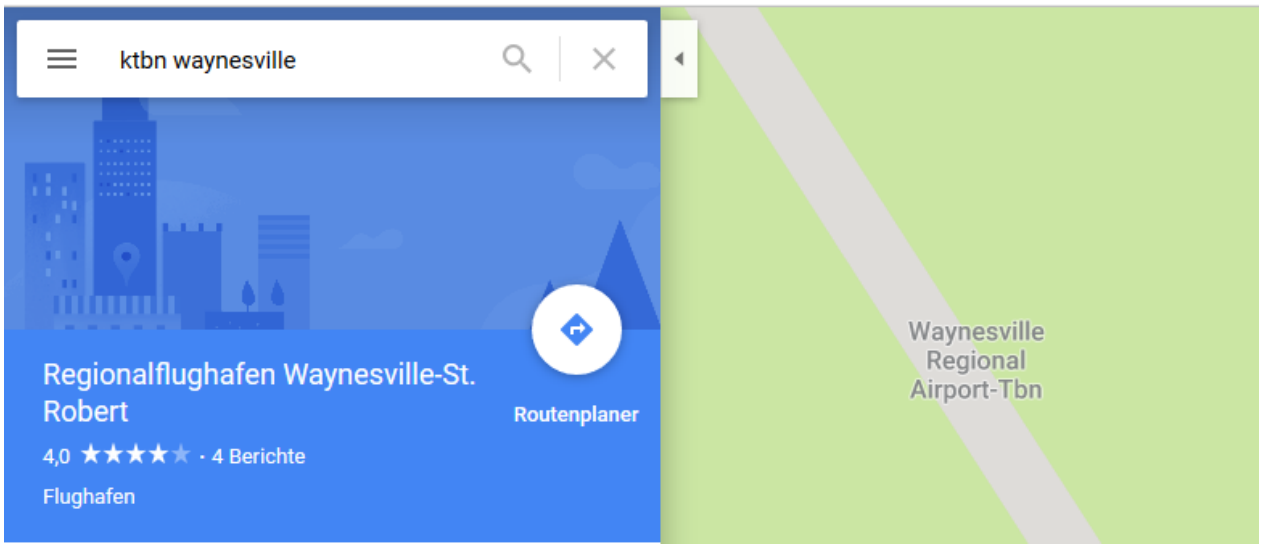
v1: 2017-04-02

Is it possible to add a custom airfield to aerofly just with AC3D, without using 3Dmax or Cinema4D? Yes.
This is probably not the 'official' way, and of course it is not perfect.
There is still plenty room for improvement, but in fact we can add an airport at a basic level very quickly.
In this tutorial I did not pay attention to runway markings, this is just a matter of bitmap textures.
So it is not a problem of the workflow, just texture work.

Please find 2 zips included:
One contains all data for the SDK environment, the second contains the final aerofly FS 2 airport.
We can really modify the existing ground mesh with a custom mesh.
This means also, we could add slope runways by defining the mesh with a slope. This will be a later task.

A. First, let's get the coordinates.
If you are not familiar with geo coordinates, please read my tutorial aerofly FS2 Rodeo Object import.pdf
https://drive.google.com/open?id=0BwGOTx-xnyyWQnlxUzJxTl9yQVU

Open Google Maps and search for ktbn Waynesville.



Switch the map display to earth (left bottom) and center the runway.
Our first step is to evaluate the length of the runway and the center for our graphics work.
Right click and select the function Measure Distance. In our example the length is 184km = 1840m.

So the center of the runway is at 920m We try to find this point and look for some characteristic features.

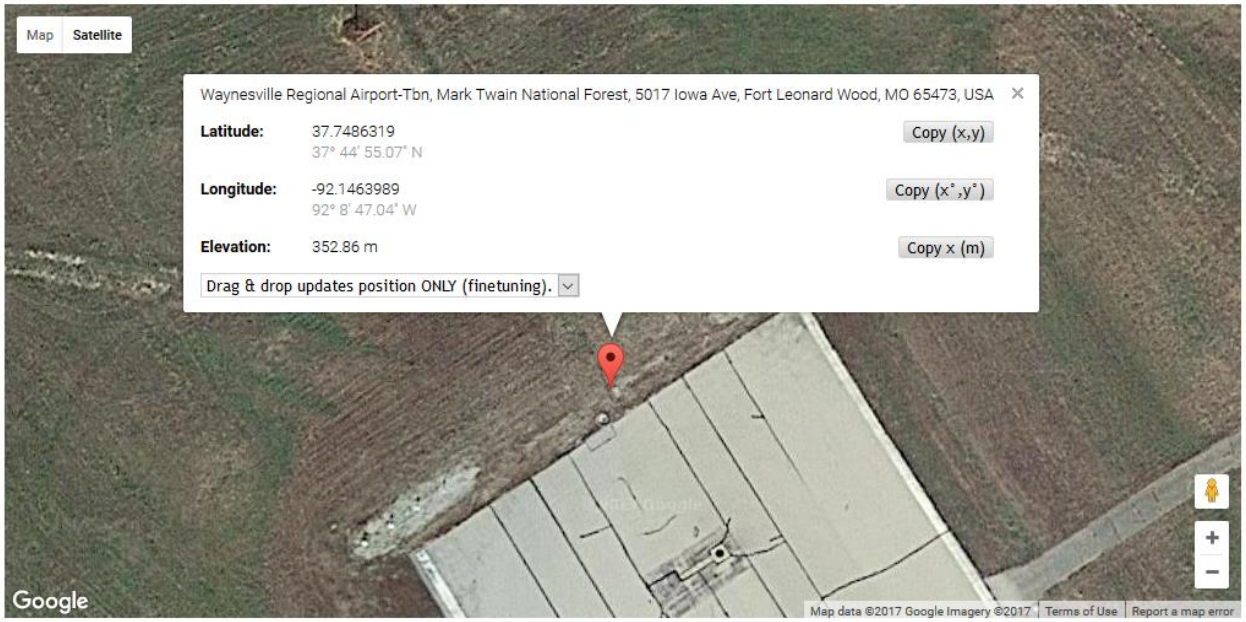
You can just move the runway endpoint backwards until you find the appropriate length.
Btw, you can zoom during these operations.

Now open the site http://en.mygeoposition.com/ and search for ktbn Waynesville.

Our target is to get the coordinates of the runway lights and the center of the runway.
Again, switch to earth display and search the points we have identified in Google Maps.

As you can see, this is exactly the same position in both applications.
These are the coordinates we will enter I our TSC file later.

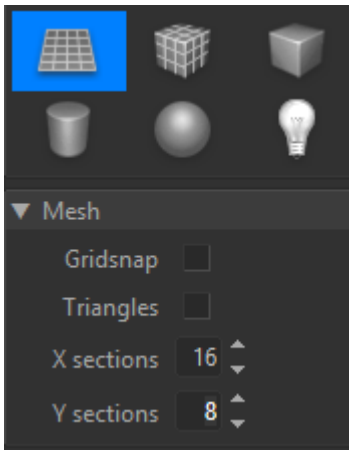
Let's find the center of the runway lights. Do this for both sides and note these coordinates as well.

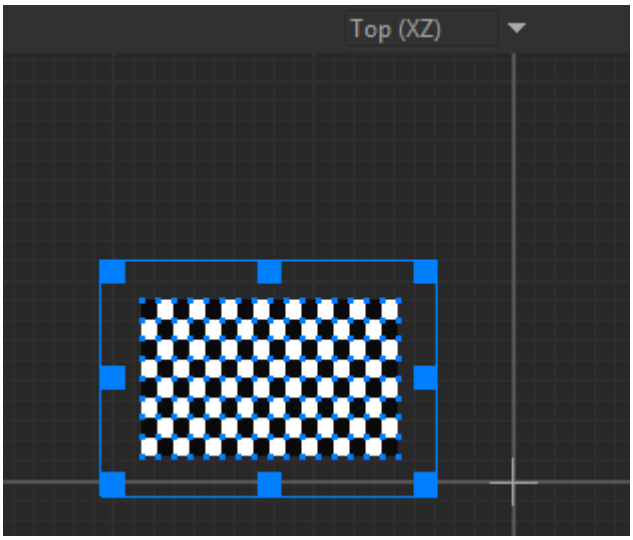


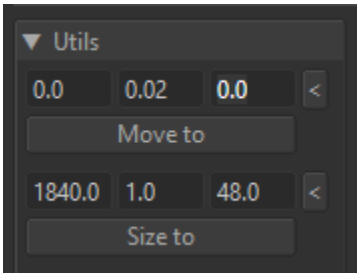
Last action here, measure the width of the runway. This is for Waynesville a width of 48m.
Note: Of course you can get these information on a different way, e.g. . https://www.airnav.com/airports/

B. OK, now we enter the graphics. Start AC3D.

Place a mesh in the top view

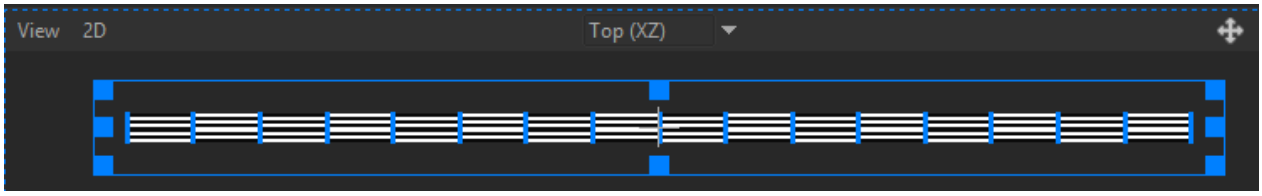




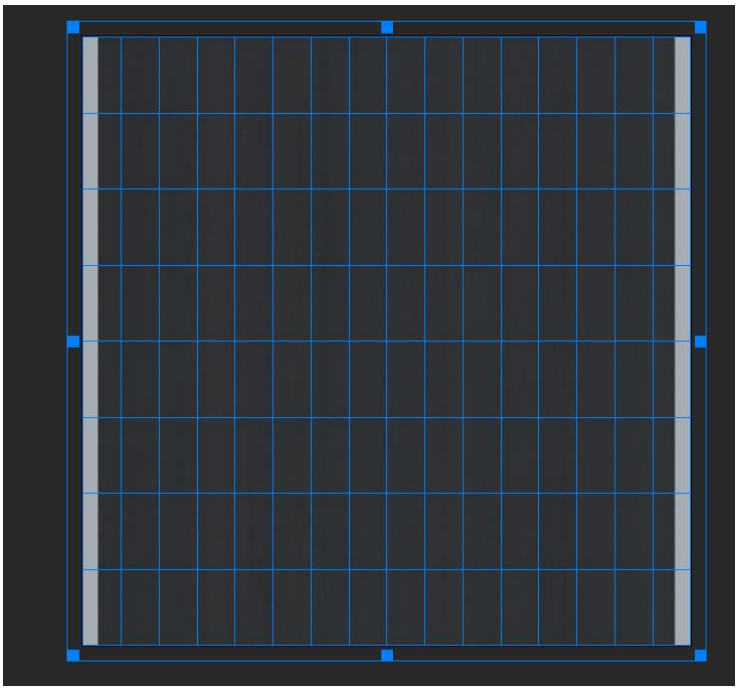


Move the mesh to position 0 by pressing Move to.
We may also add an elevation of 0.02m above 0.
Next enter the length 1840m and width 48m and press Size to.

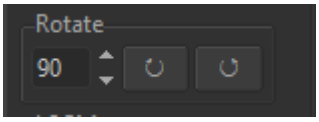
Note: The x axis is the length and this axis points to north in aerofly.
The z axis in AC3D is the width of the runway.



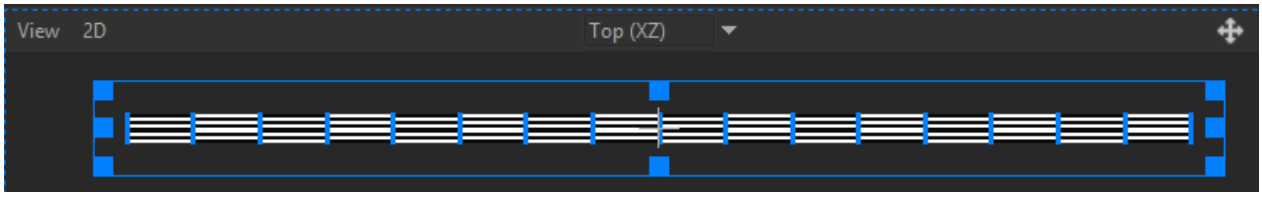
We have to place a texture onto our runway. My example file is called rwy_color.bmp.
Select All and then Object – Texture - Load texture – rwy_color.bmp



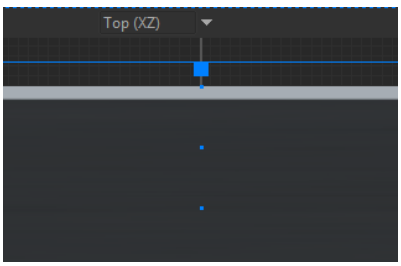
Press F10 to enter Texture Coordinate Editor.
We see, the orientation has to be rotated for 90 degrees.



Leave Texture Coordinate Editor.



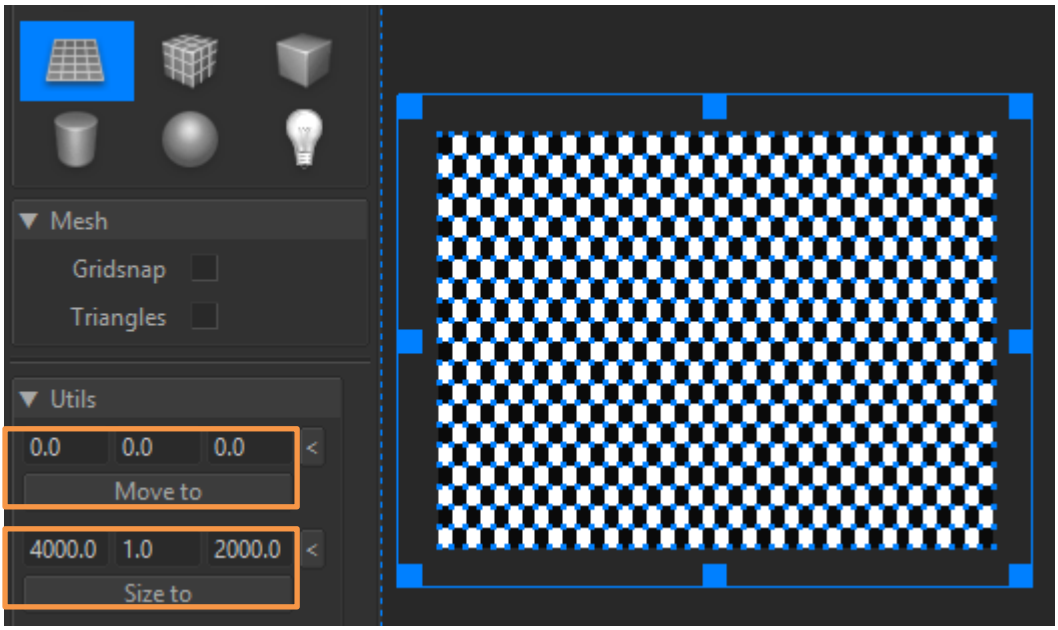
If your result shows still black and white boxes, set palette to white.



Of course, this is the moment to add custom textures, add runway signs and so on.

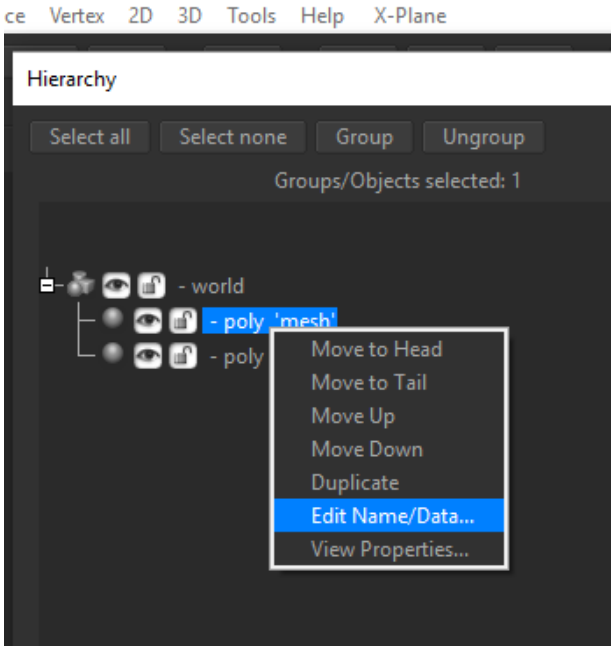
We will create another object in AC3D, this is the ground mesh.
It is impressive, that we can modify the existing ground mesh in aerofly with a custom mesh from AC3D.

We want to cover an area of 4km x 2km = 4000m x 2000m.
Create this mesh, move it to the center, modify the size to the expected values.

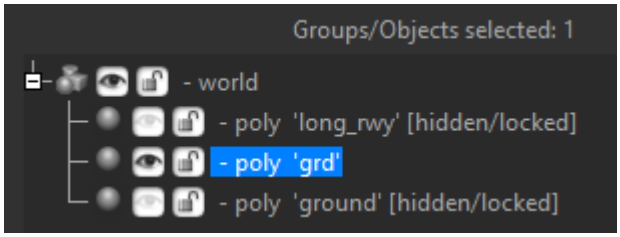


This mesh stays without texture.

Btw, we can name the layers by pressing F8. Right click the name and set the runway to long_rwy, the ground mesh to ground.



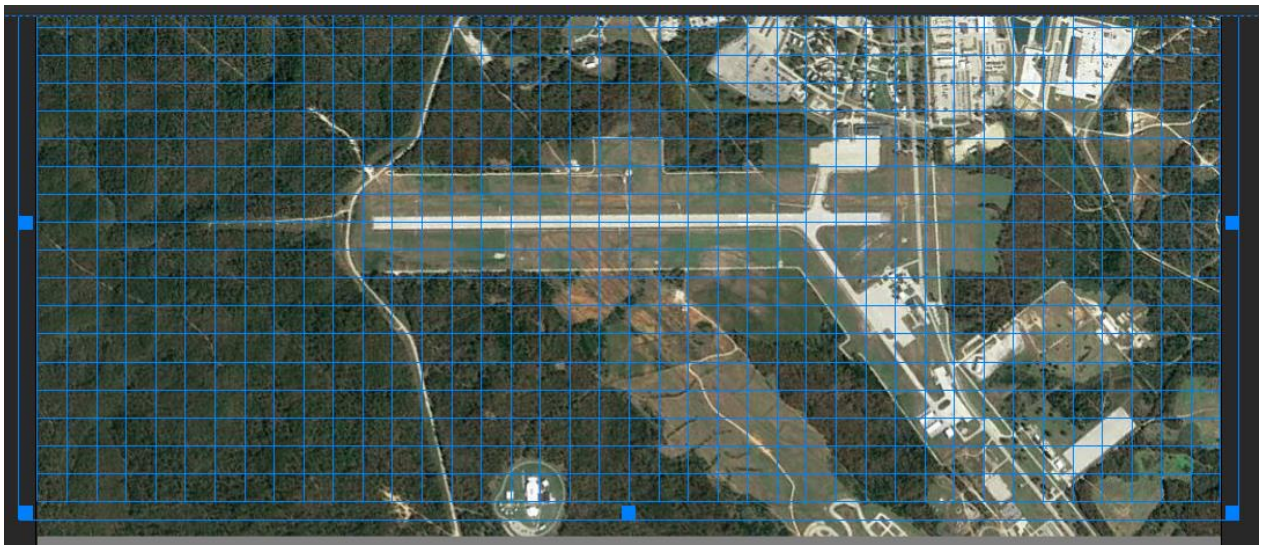
Now we duplicate the ground layer. Select the ground, Edit – Duplicate. Name the layer grd.



We will place an environment texture on it. Our example includes a grd_color.bmp,
which is a snapshot from Google Maps.
Select the mesh grd, Object – Texture - Load texture – grd_color.bmp.
The texture will be displayed at the grd mesh.

The BMP contains only half the environmental ground, the lower part is free space to add buildings textures later on. We have to shift the texture mapping, that the runway is exactly alined in the center.

Press F10 and move the lower part of the mesh upwards, until the runway is aligned with the center line.

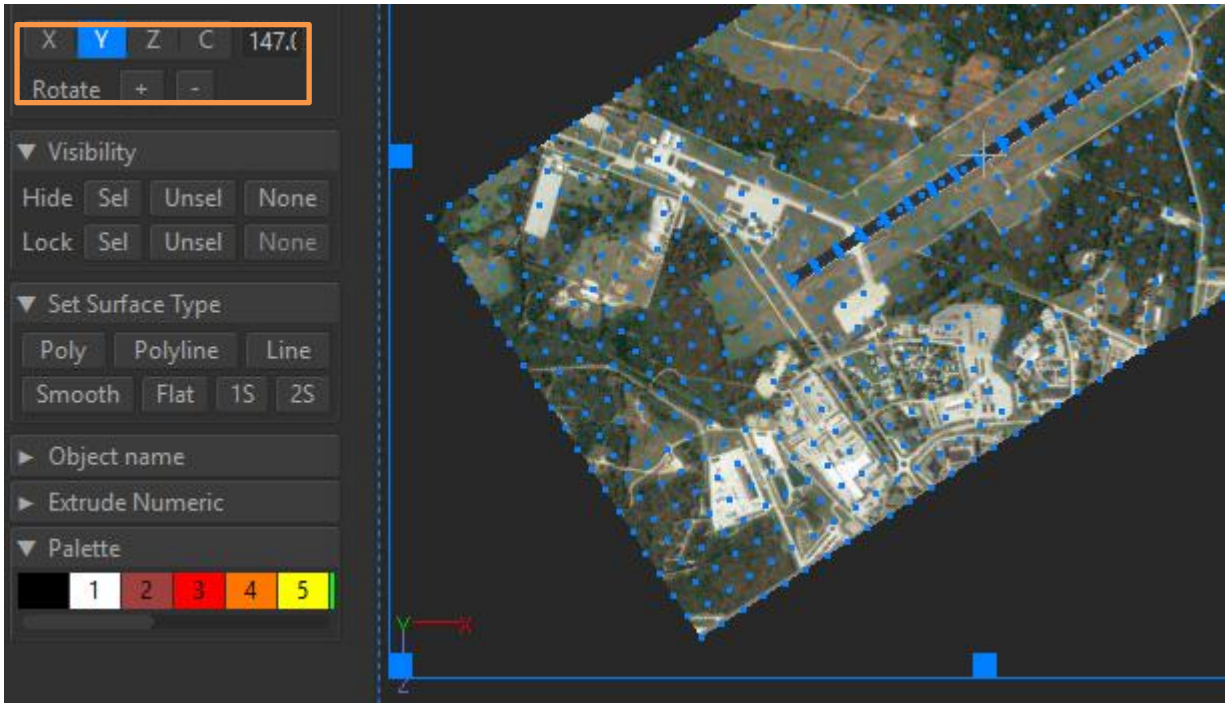


As we see a slight deviation, we can rotate it about 0.5 degrees counterclockwise.
This is the result.

Next step is to rotate the whole work to the actual heading of the runway.
We get this information from angle measuring or from airport pages like:

https://www.airnav.com/airports/ Runway heading: 144 magnetic, **147 true**

We rotate all objects along the Y axis to this angle.



That's the graphics part. We have to export 3 different TGI files with the aerofly exporter.
I save each layer as a separate AC file and use then the function File – Export- IPACS TGI files.

Please use the predefined names ground.tgi, grd.tgi, long_rwy.tgi

C. Structural work with windows explorer, notepad++ and IPACS converter.

We need the file named content_converter_config.tmc in our directory.
The only adjustment here is the name of the output folder:

<[string8][output_folder][scenery/places/ktbn_waynesville/]>

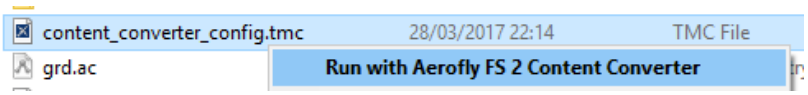
We need the file ktbn_waynesville.tsc in our directory.
Please check this file.

Attention: There are slightly different coordinates in the file compared to the screenshots above.
I did not replicate the very same number for the tutorial, but the deviation is minimal.

```
<[file][][]
<[tmsimulator_scenery_place][][]
////////////////////////////////////////////////////////////////
//
// general information
//
////////////////////////////////////////////////////////////////
<[string8]    [type]           [object]>
<[string8]    [sname]          [Waynesville]>
<[string8]    [lname]          [Waynesville Regional Airport]>
<[tmvector2d][position]        [-92.1407488 37.7416487]>
<[float64]    [size]           [5000]>
<[bool]       [autoheight]     [false]>
<[string8]    [coordinate_system][lonlat]>
////////////////////////////////////////////////////////////////
//
// objects
//
////////////////////////////////////////////////////////////////
<[tmsimulator_scenery_objecttmslist][objects][]
  <[tmsimulator_scenery_object][element][0]
    <[string8][type][ground]>
    <[string8][geometry][ground]>
    <[tmvector3d][position][-92.1407488 37.7416487 350]>
  >
  <[tmsimulator_scenery_object][element][1]
    <[string8][type][object]>
    <[string8][geometry][grd]>
    <[tmvector3d][position][-92.1407488 37.7416487 350]>
  >
  <[tmsimulator_scenery_object][element][2]
    <[string8][type][object]>
    <[string8][geometry][long_rwy]>
    <[tmvector3d][position][-92.1407488 37.7416487 350]>
  >
>
```

9: Short name (will be displayed in the Location and Navigation map of aerofly).
10: Long name (used where?)
11: The coordinates of our runway center, the center of our whole project.
12: Size 5000
13: autoheight must be set to false to use our own ground mesh height
22: type ground is important to use this as a ground mesh
23: name of the ground geometry file
24: coordinates plus an average height of our mesh. We have seen this height in en.mygeoposition.com
27: type object
28 name of the object geometry file grd
33: name of the runway geometry file long_rwy

This is enough to run the converter.
In windows explorer, right click content_converter_config.tmc and select run with Aerofly…



This process creates a new directory in …\Documents\aerofly FS 2\scenery\places\ktbn_waynesville and adds all necessary files.

If you rerun the converter, it will process only modified files.
If you want to do the full process, remove the directory.
Do not mix up this file with the same file name in our SDK environment.
Keep SDK data and final data separate!

Please use the provided TSC and compare the data.

```
1  <[file][][]
2      <[tmsimulator_scenery_place][][]
3          <[string8][type][airport]>
4          <[string8][sname][Waynesville]>
5          <[string8][lname][Waynesville Regional Airport]>
6          <[string8][icao][KTBN]>
7          <[string8][country][USA]>
8          <[string8][coordinate_system][lonlat]>
9          <[vector2_float64][position][-92.1407488 37.7416487]>
10         <[float64][height][350]>
11         <[float64][size][5000]>
12         <[vector2_float64][tower_position][-92.1389812 37.7424432]>
13         <[float64][tower_height][362]>
14         <[bool][autoheight][false]>
15         <[string8][lights][]>
```

This lists only the additional entries:
3: type airport
4: short name
5: long name
6: ICAO code
7: country
9: coordinates
10 height above sea level (used in the location and navigation menu)
11: size
12: tower camera coordinates (not necessary, but you can specify the camera view)
13: tower height above sea level (not necessary, but you can specify the camera view)
14: autoheight false

```
16          <[list_tmsimulator_scenery_object][objects][]
17              <[tmsimulator_scenery_object][element][0]
18                  <[string8][type][ground]>
19                  <[string8][geometry][ground]>
20                  <[vector3_float64][position][-92.1407488 37.7416487 350]>
21              >
22              <[tmsimulator_scenery_object][element][1]
23                  <[string8][type][object]>
24                  <[string8][geometry][grd]>
25                  <[vector3_float64][position][-92.1407488 37.7416487 350]>
26              >
27              <[tmsimulator_scenery_object][element][3]
28                  <[string8][type][decal]>
29                  <[string8][geometry][long_rwy]>
30                  <[vector3_float64][position][-92.1407488 37.7416487 350]>
31              >
32          >
```

18: type ground (important for our mesh)
19: geometry name ground
20: always the same coordinates, additional height (important for the mesh height!)

23: type object
24: geometry name grd

28: type decal (avoids flickering of the runway above the ground texture)
29: geometry name long_rwy

```
33          <[list_tmsimulator_scenery_object_animated][objects_animated][]
34          >
35          <[list_tmsimulator_runway][runways][]
36              <[tmsimulator_runway][element][0]
37                  <[tmvector2d][endpoint1][-92.1464029 37.7486032]>
38                  <[tmvector2d][endpoint2][-92.1350679 37.7346755]>
39                  <[tmvector2d][threshold1][-92.1464029 37.7486032]>
40                  <[tmvector2d][threshold2][-92.1350679 37.7346755]>
41                  <[float64][width][48.00]>
42                  <[string8][name1][14]>
43                  <[string8][name2][32]>
44                  <[string8] [appltsys1][malsr]>
45                  <[string8] [appltsys2][malsr]>
46                  <[string8] [papi1][left]>
47                  <[string8] [papi2][left]>
48                  <[string8] [reil1][none]>
49                  <[string8] [reil2][none]>
50              >
51          >
```

37: center point at one end of the runway
38: center point at the other end of the runway
39: center point of the threshold light position (can be identical with runway endpoint)
40: center point of the other threshold light position (can be identical with runway endpoint)
41: runway width
42: runway number
43: runway number opposite direction
44: name of approach light system 1 (
45: name of approach light system 2
46: papi1 position (left of runway)
47: papi2 position (left of runway)

```
<[list_tmsimulator_startposition][start_positions][]
>
<[list_tmsimulator_parking_position][parking_positions][]
    <[tmsimulator_parking_position][element][0]
        <[tmvector2d][position][-92.1353053 37.7368603]>
        <[float64][heading][260.00]>
        <[string8][name][Parking Position #1]>
    >
>
```

We may add start or parking positions: Just coordinates and heading.
This is the choice when selecting the airport in Location or navigation menu.

Note: IPACS has a structure organisation for the airports in the original places folder:
They use a subfolder for the countries like usa, ch,
and another subfolder for the us states.

We can duplicate this structure and move the final folder into this structure:
…\Documents\aerofly FS 2\scenery\places\usa\mo\ktbn_waynesville

We can add buildings and static elements to this TSC as well.
Please refer to my tutorial aerofly FS2 Rodeo Object import mentioned already at the first page.

That's it for the moment.
IPACS will change some parameters to make it even easier for the developers.
So this tutorial may change in a near future. I will continue to update the version.

Hope you enjoy this tutorial.
Karl-Heinz (Rodeo)