Introduction to MSFS SDK

How to edit & create your own airports

Authors: Seb Brown & Dragoljub Blagojevic

Contact: drago@flightsimsweden.se

Revision: 22 august, 2020

Table of contents

Introduction

MSFS SDK: The SDK is still not nearly finished which means that information below can become obsolete at any point in time. If you take and read the SDK, in its current state it only explains how to do things, it fails to explain how to get there, which is where this tutorial comes in.

The way Microsoft has made the developing work is with a similar process to that of the ESP, where one was supposed to create own bgl files and simply keep the actual released scenery files separate from those that you as a developer compile. This makes the development currently a bit of a pain but it's what it is so lets get down to it.

Before we dig in, make sure that you have enabled the developer profile within *general* → *Developers* setting inside your simulator..

If at any point you come across a problem, feel free to drop a message on this thread, and we will respond to it as soon as possible.

Folder structure of your project (method by Lyriax)

(NOTE: If you wish to add custom objects in your airport, skip straight to the “Adding Custom Objects Via Blender” section, as it requires a different structure.)

Creating a proper folder structure at the start of the project is *vital*. Go to a place in your ***windows explorer (WE)*** and create a folder, where you can choose to name it. For the purpose of this tutorial I am naming mine dsc (for my brand), underscore esnb (icao) underscore sollefteå (town): dsc_esnb_solleftea. This folder is placed within Documents/MyFSProjects on my C drive but you can really place it anywhere you want.

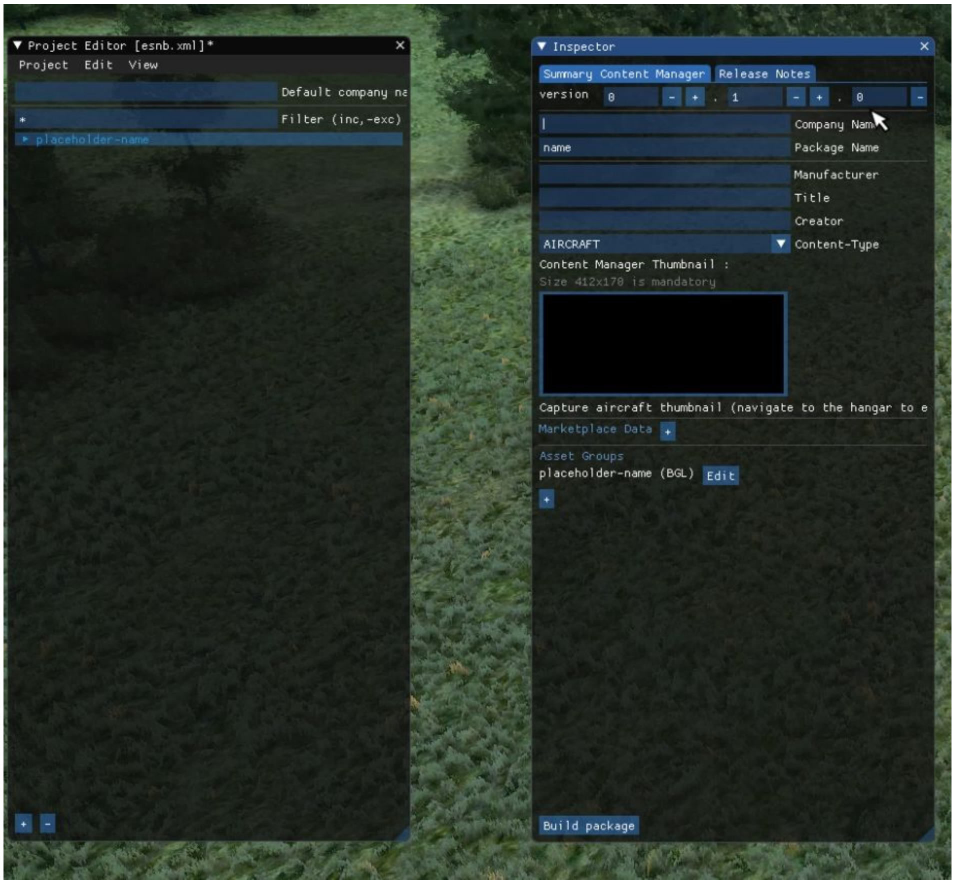
Create two folders within your project called *scene* and *scenery*. Within the scenery folder, create a folder called *world.* Inside the world folder create a folder called *scenery*.

You are now ready to commence building/editing your airport!

Creating a new airport (method by Lyriax)

Grab coordinates somewhere of the location you wish to build your scenery in a format of decimal degrees (DD) like so: 41.40338, 2.17403. When you are about to load a flight insert these within the search bar on the top left side. You will then be presented with an option in H:M:S format. Click on it and you should then be taken to the location of your choice.

Once spawned, go to camera and enable *Developer Camera*. Go to *DevMode* and click on *New Project*. Using the window in front, navigate to the folder of your project and once inside, click on *Select Folder*. In my case it is Documents/MyFSProjects/dsc_esnb_solleftea. Under the *Project Name* type in the icao code of the airport. Make sure you press *ENTER*. Click on *OK*. If you have not done it yet, open *Project Editor* under *Tools* and click on the + button at the bottom-left of it's window. For the type, choose *BGL* and name it *placeholder-name*. Remember to press *ENTER*. Click *Create*. Under tools, open up *Inspector*. Your workspace should now look something like so



```
*                                   Filter (inc,-exc)
▸ placeholder-name

version  0 . 1 . 0
name                                Package Name
AIRCRAFT ▼                          Content-Type
placeholder-name (BGL)
```

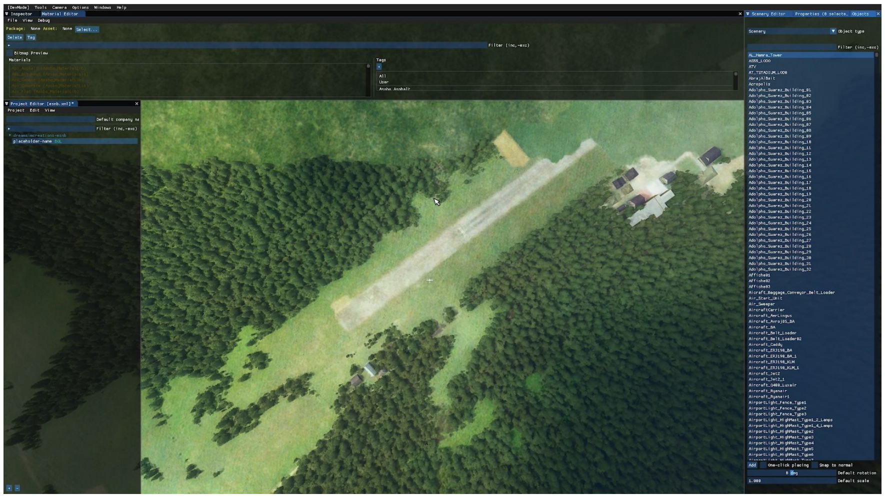
Creating an Airport (cont.)

Under *Company Name* enter your brand/company name. Under *Package Name* enter your airfields icao code. Let me yet again remind you of the need of pressing *ENTER*! Under *Content type* choose *Scenery*. Click on the thumbnail and go to the place where you installed your SDK, in my case MyDrive/MSFSSDK and navigate to *MSFSSDK\Samples\SimpleAirport\PackageDefinitions\mycompany-airport\ContentInfo*. Inside you will find a generic thumbnail. Select it and click on *Open*. Under *Asset Groups* inside the inspector window find the *Edit* button. It should be next to the text called *placeholder-name*. The *Asset Directory* inside the inspector should now be navigated to the projects subfolder called *scene*. The *Output Directory* should be however typed in manually to display *scenery\world\scenery*. Make sure to press *ENTER*. Now press *Load In Editor* which should open the *scenery editor* and move your dev camera to a top-down view.

Now is the proper time to set up your work space! Under tools make sure that following is displayed in front:

- Project Editor
- Material Editor
- Scenery Editor
- Inspector
- Properties (to bring this up right click anywhere and bring up the window).
- Objects window

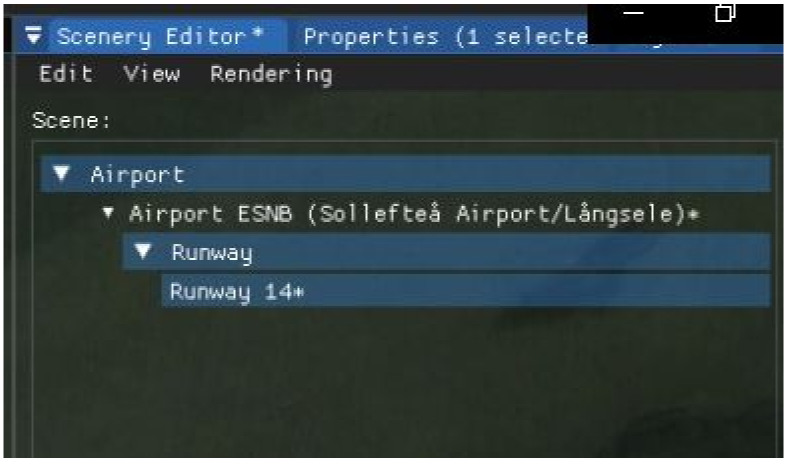
Drag the window you wish to “dock” to the workspace by dragging it to one of the blue-boxes on each side. You can dock multiple windows on one button. Dock all the windows so that the end-result looks something like so



Creating an Airport (cont.)

The first thing we need to do now is add an Airport instance to the project! Go to *Objects* and choose *Airport* under Object type. Click on *Add*. Head over to *Scenery Editor* and you should see an Airport instance there. Once confirmed, head over to the *Properties window* and add ICAO and the airport name. If it's a new airport (and not there by default), bock in *Apply Flatten* and *Star airport*[1]. Head over to *Project Editor* and click on *Save*. Do the same in the *Scenery Editor* but save it inside the ***scene*** folder located within your project directory, in my case *Documents/MyFSprojects/dsc_esnb_solleftea/scene*. Save it as *airport-icao*, in my case airport-esnb. Save it.

To create a spawning point, you can just create a simple runway. Click on the *Objects* window and select a *Runway*. Click now *Add* without any other options bocked in. Go to *Scenery Editor* and under the *Runway* tab, click on the runway option, in my case *Runway 14+*.



In front of you should be a rectangle with pivoting points. The one located in the cross section is for the length of the runway, the ones at the edges are for the runway rotation. You can also set the runway config in *Runway configuration*. If the runway seems to look like its under the mesh, disregard - its just a bug in the scenery maker.

[1] Star airport gives the starmarking that handcrafted airports have on the world map so that it's easier to find.

Overwriting An Existing Airport

If the airport you are making already exists in the simulator, you may wish to overwrite it. By doing this, it will act as if the default airport never existed, and will be replaced by your customized scenery.

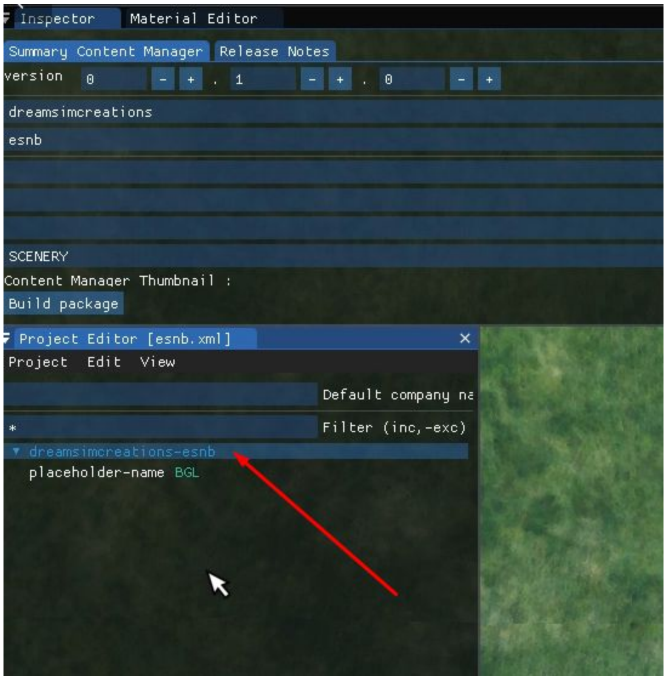
1. Select the airport, and open the properties page.
2. Under the “To Delete” tab, tick all of the boxes.
3. Save the scenery.

You can then build the scenery and reload to see the effect.

Compiling an airport (method by Lyriax)

Now that we have a runway (read: a spawning point), we need to compile the airport so that you don't have to go through all the hassle of finding the airport again.

Go to *Inspector* and select the main project



version 0 . 1 . 0

dreamsimcreations

esnb

SCENERY

Content Manager Thumbnail :

Default company na

* Filter (inc,-exc)

▾ dreamsimcreations-esnb

placeholder-name BGL

Under the inspector, select *Build Project*. Console window should be open now. Disregard all the errors and check within windows explorer that the project was correctly compiled within a subfolder called *Packages*. The folder within is the folder that goes inside the community folder of MSFS and is the folder that needs to be shared. Now, place this folder within your own MSFS\Community folder. If you wish to check any edits you have done you have to close your MSFS, update the scenery that you placed earlier within community folder and then start MSFS.

Creating an exclusion zone

An exclusion zone is a zone in which underlying layer-stuff like buildings and trees are excluded. To add an exclusion zone head over to the *Object* window and select *Polygon*. Click on *Add*. Now head back to *Scenery Editor* and look for a red Polygon tab. Doubleclick on the red. To start placing the polygon hold down CTRL + LEFT MOUSE click to create a node. To close the polygon press *ENTER*. Go to the *Properties* to select what you want the polygon to exclude. You can even make the polygon filled with trees, asphalt and other stuff.

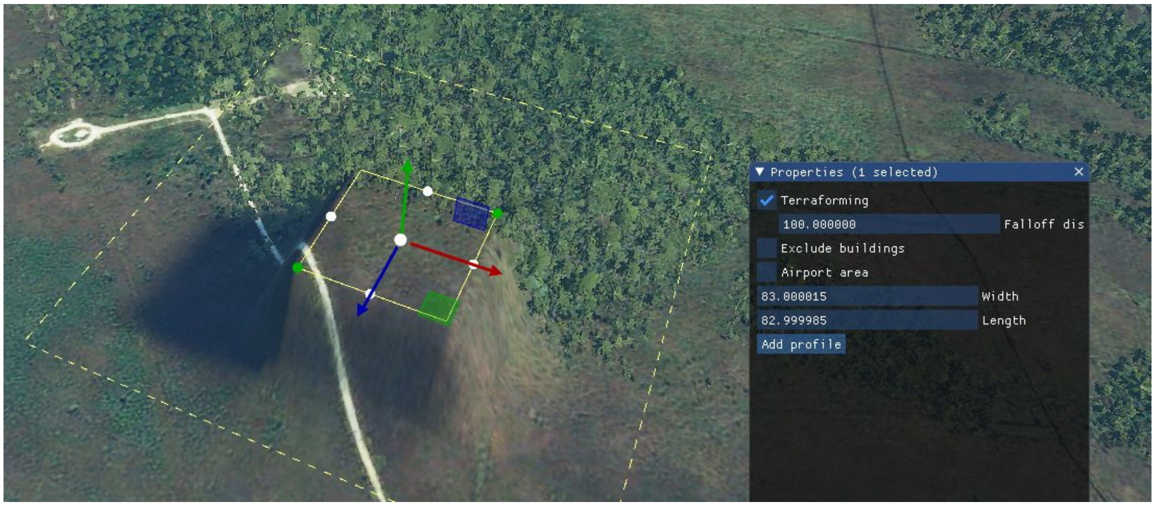
Flattening An Airport

Some airports or areas in MSFS have unnecessary hills, which can be fixed by flattening the airport.

1. Inside the properties of the airport, tick the “Always Flatten” box.
2. Create a polygon, and select the outline of your airport (CTRL + LMB to add a node and ENTER to finish the selection).
3. Inside the properties of the polygon, select “Terraforming” and select “Airport Area”.

Terraforming terrain

Terraforming terrain is a vital functionality of the SDK that will provide you, the developer with the ability to mold the terrain to your liking. To terraform a terrain you select a rectangle object within *Objects* tab, and under *Properties* bock in *Terraforming* option.



▼ Properties (1 selected) ✕

☑ Terraforming

100.000000 Falloff dis

□ Exclude buildings

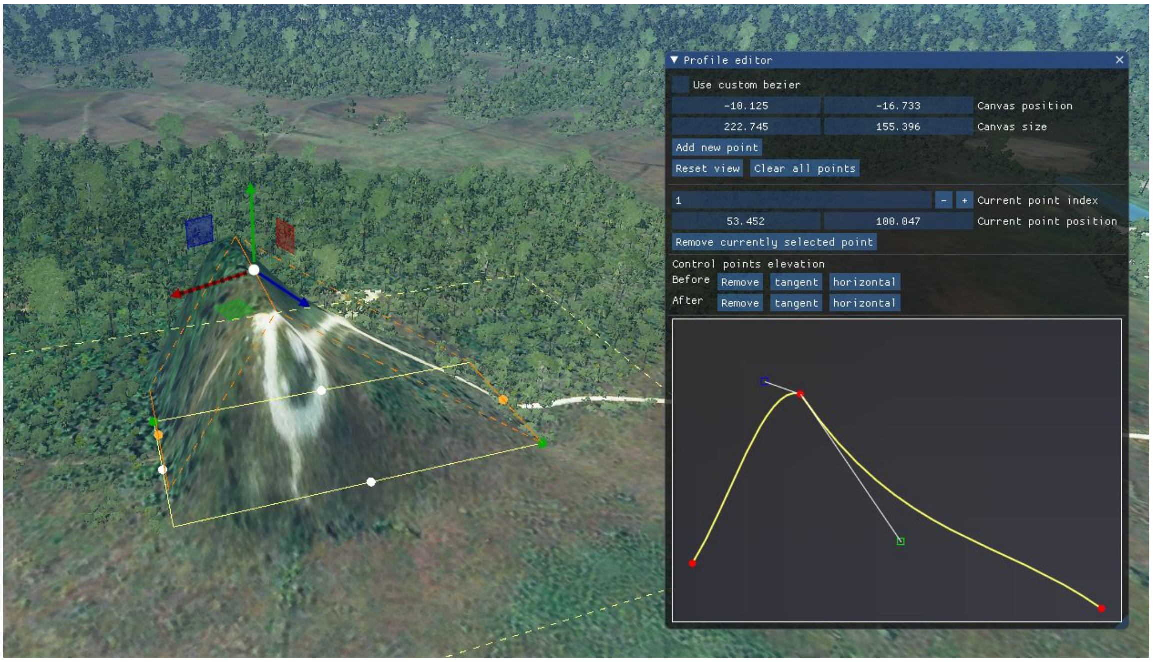
□ Airport area

83.000015 Width

82.999985 Length

Add profile

Second step is to open the *Profile Editor* to edit the general slope among other things of the rectangle profile.



▼ Profile editor ✕

□ Use custom bezier

-10.125 -16.733 Canvas position

222.745 155.396 Canvas size

Add new point

Reset view Clear all points

1 - + Current point index

53.452 100.047 Current point position

Remove currently selected point

Control points elevation

Before Remove tangent horizontal

After Remove tangent horizontal

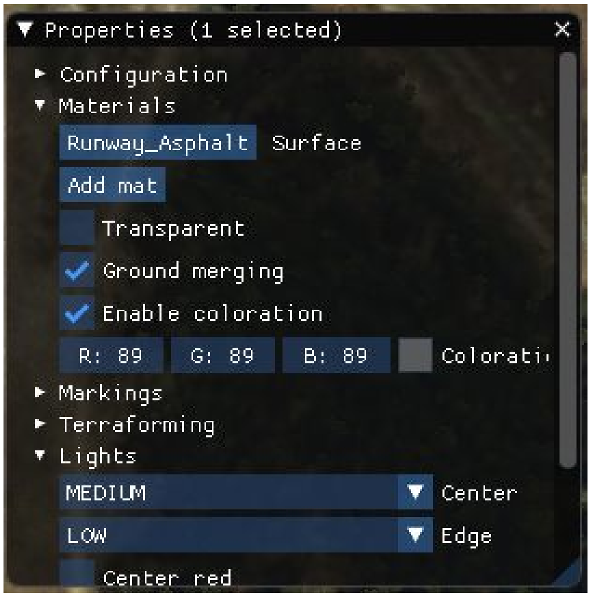
Changing apron and runway textures

Let’s say you’ve made an apron for your airport, but the texture and color doesn’t match that of the satellite imagery. Thankfully, Asobo Studios has made a simple way to change this in a matter of clicks.

1. Make sure that you first have your surface (in my case a runway) selected and the properties window open, if you do not have this window open, right click on the surface and make sure that the “Properties” box is ticked. Now is also a good time to open the “Material Editor”, which you should have docked at the top along with the inspector.
2. For runways, the material will show inside the “Materials” tab in properties. For aprons, the material will show in the first section of properties. If no material shows, you may have to click “Add mat” to add one, but the steps after that are the same.
3. Inside the material editor, there are two sides; the left hand side shows a list of specific materials, and the right hand side shows a list of surface types. For me, the runway is Asphalt, so I’ll select “Asobo_Asphalt”. You should see that it updates the list on the left. You can now look through the list (the left one) and find a material that best matches your actual airport. If you wish to see a preview of it, tick “Bitmap Preview”, which is located directly above the left hand list, now when you hover over the items, it should show two images, an albedo and a normal map; for this, only look at the albedo map, which is the one on the left hand side.
4. Once you’ve found your material, make sure you have it selected on the list, you can tell when it’s selected by the line it is on being blue instead of black. Once it’s selected, you can click on the box beside the word “Surface” inside properties. This should overwrite the old texture and replace it with the new one. Please note that you may have to open the scenery editor again for the change to take place.

Changing apron and runway textures (cont.)

Another thing the SDK allows you to do is customise the color of your material. For this, select “Enable Coloration”, which you’ll find just below where you set the material. Once you’ve done this, you can either set the values manually, with RGB values, or you can click on the right most box and use a color picker from there. Please note that you may have to open the scenery editor again for the change to take place.



▼ Properties (1 selected)

▸ Configuration

▾ Materials

Runway_Asphalt Surface

Add mat

□ Transparent

☑ Ground merging

☑ Enable coloration

R: 89 G: 89 B: 89 ■ Colorati

▸ Markings

▸ Terraforming

▾ Lights

MEDIUM ▼ Center

LOW ▼ Edge

□ Center red

Utilizing default assets

Another thing you may want to do with your airport is add default assets; these can bring life to your airport with vehicles, vegetation and smaller buildings. The first thing you should do is remove the pre-existing default assets already at the airport, if your airport already existed.

Overwriting Default Airport Objects

1. Add a polygon. In the objects tab, select “Polygon” and press add. You should now note that in the scenery editor, the polygon does not lie under the airport; because of the way Asobo made the SDK, the polygon files must be saved separate from the airport. Before saving, we must place the polygon first.
2. By double clicking on the polygon, you should get a top down view, using this view, align yourselves to the area with buildings you want to remove.
3. Press CTRL + LMB to add nodes, and once you have outlined the area, press enter.
4. Inside the properties for the polygon (right click, make sure properties is ticked), you will see different options that are subset after “Exclude All”. By pressing “Exclude All”, your polygon will remove everything inside it, including buildings and vegetation. If you just want to exclude buildings, tick “Exclude OSM Buildings”. You may have to try a number of different options, as it may depend on what type of airport you are making.
5. Once you’ve done that, in the scenery editor, press “Save Scenery”. It will once again prompt you with a file box, in which you must save the file in the Scene folder we made earlier, with the name “airport-icao-poly”, replacing “icao” with your airports ICAO code. Make sure that you did not save it with the same name as your airport, or else you will run into many problems.

Once you’ve done this, you can add as many objects as you want following the steps below.

Adding Default Objects

1. Head back over to objects, select “SimObject” as your type. It should load in a list of all SimObjects inside Microsoft Flight Simulator. For this example, we will place down a Catering Truck.

2. Search for the object you want, in my case I will search for “Catering”, and find one that sounds best. At the bottom of the objects tab, press “One-click placing”. This will allow you to place as many as you want where you want and quickly.
3. After you do this, you should see that the object you selected is now following your cursor. To place it, just left click where you want it to go. As soon as you’ve place as many as you’re happy with, untick “One-click placing”, this will stop many issues in the next few steps.
4. You can now use the gizmo to rotate and move your object, and press “Save Scenery” once you are finished.

Gizmo Controls

Move: W
Scale: R
Rotate: E

Adding Vegetation

Let’s say you have added objects to your airport, but there’s some areas where it looks as though the AI has missed some trees, luckily, Microsoft Flight Simulator’s SDK comes with a handy way to add trees to your airport.

Utilizing default assets (cont.)

Here you can see a spot where there should be trees, but they are missing. There are two ways to add trees, via a polygon, and via manual placing. For this patch, I will use a bit of both.

Adding Trees Via Polygon

1. To add trees using a polygon, you must first add one. Head over to the objects tab, select "Polygon" and press Add. When you go back to your scenery editor, you should see a polygon, which will be highlighted in red (this happens because it has no shape).
2. To place it, press CTRL + LMB to add nodes, and press ENTER to complete the operation. Once you've highlighted the area of trees, you're ready to move onto the next step.
3. Open the properties window of your polygon (you may have to right click and tick the properties box), and at the bottom, there will be a tickbox with the message "Vegetation". Ticking this will fill the polygon with trees and bushes matching the surrounding biome, another new feature to MSFS. It will also introduce several sliders, with each being able to control different features of the vegetation, such as the density and scale.
4. If the biome is not what you wish for any reason, you can press on the box beside "Select biome override" and choose a biome from there. You will then press "Save Scenery" in the scenery editor, and if you haven't saved a polygon already, and are prompted with a file box, save it in the "Scene" folder with the name "airport-icao-poly", replacing "icao" with your airport's ICAO code.

If you wish to add trees via manual placement, follow the below steps.

Adding Trees Via Manual Placement

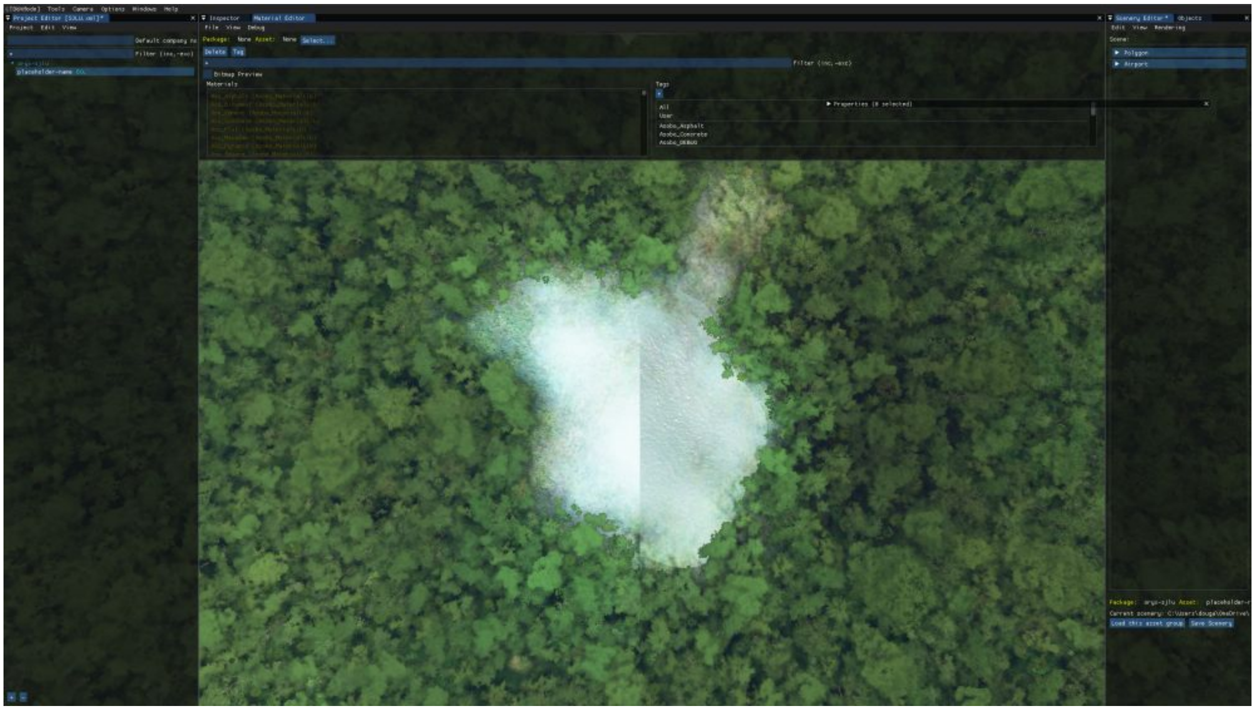
1. In the object editor, select "Scenery" in the drop down box and search for "Tree". You will note that there is a smaller selection than via the polygon method.
2. Enable "One-click placing", and add the trees by pressing your left mouse. Once this is done, disable "One-click placing" and save your scenery.

The final result should look both natural and realistic, due to the power of the MSFS SDK.

Adding Water

Another issue you may encounter with scenery is where the AI has missed a body of water. Whether this be a pond or a lake, the MSFS SDK allows you to create water via polygons.

1. In the objects tab, select type "Polygon" and press Add. Back in the scenery editor, select the polygon and use CTRL + LMB to add a node, and ENTER to complete the operation. Do this until you have highlighted your water area.
2. Once you've selected your water area, you can open the properties of the polygon (right click, make sure properties is ticked). Inside the properties, tick the box that says "Water" and select what body of water it is. By default, the water will be at sea level, to fix this, also tick the "Terraforming" box at the top of the properties panel; this will also flatten the water area, removing the 'water hills' you sometimes see in default scenery. You will then press "Save Scenery" in the scenery editor, and if you haven't saved a polygon already, and are prompted with a file box, save it in the "Scene" folder with the name "airport-icao-poly", replacing "icao" with your airport's ICAO code. That is then your completed water.



Adding Custom Objects Via Blender

(NOTE: This is not a Blender tutorial, and this process requires an intermediate understanding of Blender 2.9+. It also requires a decent understanding of the MSFS SDK basics, which you can learn about above.)

File Layout

For ease of use, I have created a template for the file layout of a scenery with custom objects. In this template, all you have to do is replace every word saying 'ICAO' with your airport's ICAO code. This includes the inside of the two XML files, inside "PackageDefinitions" and in the main folder. This is the link to the template.

Installing Blender2MSFS

In order to convert Blender objects to the format that MSFS uses for scenery. It is quite easy to install, but it requires a basic understanding of installing Blender plugins.

1. Go to this link and download the zip file.
2. Install the plugin as you would a normal Blender plugin via the zip file, not the Python file.

Exporting Your Object

The process of exporting an object is a long one, and may take a while to learn. If you have any questions, feel free to ask them on this thread.

PBR Materials

Using PBR (Physically Based Rendering) textures with Microsoft Flight Simulator requires Substance Painter, as the format for some of the textures is very hard to make by hand. The export-preset for MSFS can be found here.

Material Setup And Export

Once you've got your textures and your model, and you have the Blender2MSFS plugin installed, save your .blend file to a new folder inside 'PackageSources/modelLib' named what your object is, in my case "DemonstrationObject". The .blend file should be called the same. At the same time, move your texture files into the newly created folder. After this, you should be good for material setup.

1. Inside the materials properties, open the ‘MSFS Material Params’ tab, with your material selected. Inside of this, with the dropdown box, select ‘MSFS Standard’, and select the correct textures for the correct type. The Substance exporter gives you an Albedo texture, a Comp texture and a Normal texture. The albedo goes in the first image box, comp in the second and normal in the third. Do this for all of your materials.
2. File > Export > Extended gLTF 2.0
3. Inside the box that opens when you do this, go to the top right and beside ‘Format’, select the last in the dropdown, which should be ‘gLTF Separate (.gltf + .bin + textures)’. In the box beside ‘Textures’, paste the following minus the quotation marks: “../texture”. Tick the ‘Remember Export Settings’ box so you don’t have to enter all of this again. Under the MSFS drop down tab, tick the ‘Generate/Append XML’ box, and name the XML file the same as what you are naming your object, in my case, ‘DemonstrationObject’. Also tick the ‘Generate GUID’ box, as this will come in handy later. Under the ‘Include’ drop down tab, tick the top two boxes.
4. Press ‘Export’. This may take a while.
5. Inside your folder, you should now see a few new files. Open the new XML file that was created with your gLTF, and copy the number you see in there, that’s your GUID. Paste it somewhere where you can find it later on.
6. Create a new XML file called ‘Placeholder’ and paste this inside:

    ```
    <?xml version=”1.0” encoding=”utf-8” ?>
    <ModelInfo version=”1.1” guid=”{YOUR GUID}”>
    </ModelInfo>
    ```

7. Replace ‘YOUR GUID’ with the number you got earlier.

8. Delete the old XML file and rename the new one you just created to the same name as your old one, which should be the same name as the gLTF file.

Placing It Inside The Simulator

To place your object inside the simulator, follow the steps below:

1. Open the simulator, and open the main project file in the project editor (It will be the first XML you see when you open the airport files). Inside the inspector, with the package selected press “Build Package”. (See ‘Compiling An Airport’).
2. Close and reopen the project (close the project and the project editor, then reopen), and open the package again. This time, select the BGL file and in the inspector press ‘Load In Editor’. This will open the Scenery Editor.

3. Open the ‘Objects’ tab (View > Objects) and select type ‘Scenery’. Wait for the objects to load in and search for your object (the same name as the gLTF and the XML). Place it as normal and use the gizmo to move, rotate and scale it.
4. To save the airport, press ‘Save Scenery’ and save it inside “PackageSources/scene” the same way we saved an airport earlier in the docs.

Congratulations, you should now have been able to place your own custom objects inside Microsoft Flight Simulator.