

One-click hangars

One-click hangars is a plugin for Trimble SketchUp (previously Google SketchUp), designed to create relatively complex and customisable hangars to use in Microsoft Flight Simulator X.

Despite the name, it does actually take a bit more than just a click to create the hangars; however, it's possible to create a relatively realistic representation of a small airfield in just a few minutes.

Installation

The zip archives you have downloaded contain plugin file (hangar.rb) and several texture files.

All three zip files are required to get the plugin to work properly.

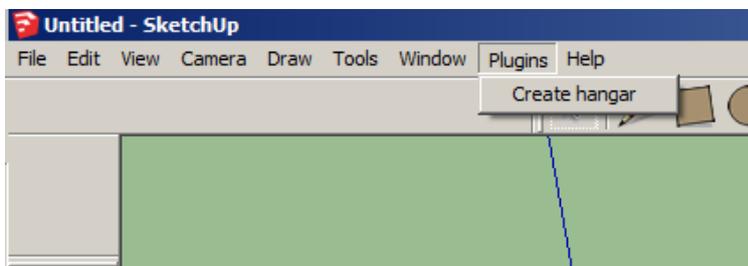
Copy hangar.rb to the SketchUp plugin folder (usually C:\Program Files (x86)\Google\Google SketchUp 8\Plugins or C:\Program Files\Google\Google SketchUp 8\Plugins)

If you haven't already, create a subfolder called \TextureFiles\ in the plugin folder and copy all jpg-files to this folder. The texture files must be in this folder for the plugin to work properly.

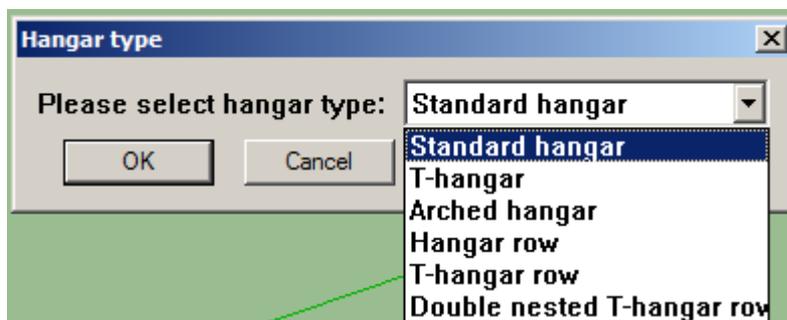
The plugin should be available next time you start SketchUp.

Interface

To create a new hangar, go to Plugins—Create hangar.



Select the desired hangar type and click OK.



There are six different hangar types to choose from, all of which are customisable. It's possible to add doors to all of the hangar types, apart from the T-hangar, and it is possible to add a range of extensions to the Standard hangar.

After selecting a hangar you will be asked to fill in several values for the hangar, before you are asked to select the textures to be used for the walls, roof, floor, etc.

Beam type?	I
Number of beams?	4
Space between beams?	500
Beam width?	20
Beam thickness?	4
Wall thickness?	10
Wall type?	Flat
Internal height?	300
Internal width?	1531
Arc height?	120
Door width?	1000
Door height?	280
Door type?	No doors
Door folding level?	20
Angle?	0
X-pos?	0.0
Y-pos?	0.0
Extensions?	

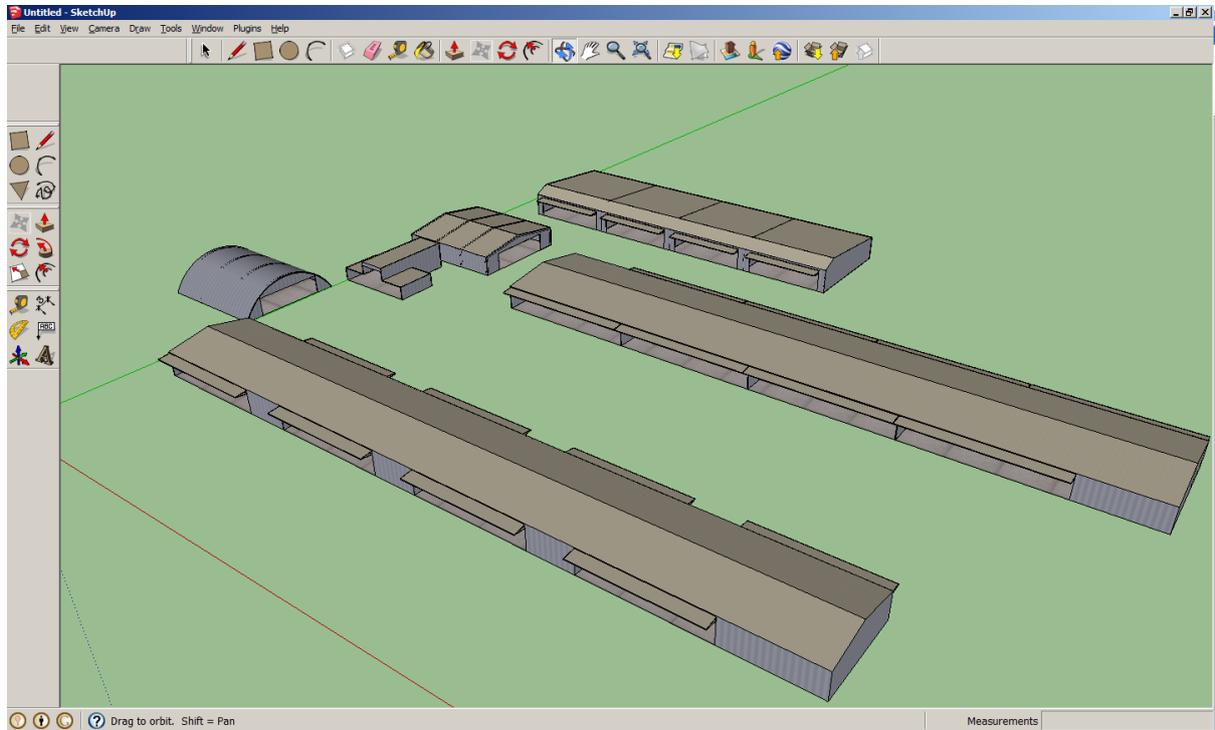
This is the hangar information input box for the standard hangar. The input box for each hangar is different, but there are some fields which are common for all hangar types. Note the Extensions field. This makes it possible to create extensions to the standard hangar.

Beam texture?	Blue metal
Wall texture?	Vertical corrugated iron
Roof texture?	Vertical square iron
Door texture?	Horizontal square iron
Door frame texture?	Silver metal
Floor texture?	Concrete

The texture options for the standard hangar. This input box is again different for each hangar type, although they all have wall, roof and floor textures.

Hangar types

The plugin can create six different hangar types; standard, t-hangar, arched hangar, row of hangars, row of t-hangars and double nested t-hangar row. The t-hangar types are very common in the US and Canada, and especially at smaller airports.



Screenshot of the six different hangar types, all created without changing any of the default values. Clockwise from the top; hangar row, double nested t-hangar row, t-hangar row, arched hangar, t-hangar, standard hangar.

Standard

The standard hangar is the simplest of the hangars. It's a rectangular hangar with a door opening in the front, although you create endless variety of standard hangars by changing the various values.

T-hangar

The T-hangar is slightly more compact than a standard hangar, and there are fewer values to change.

Arched hangar

This is a standalone arched hangar.

Hangar row

This is a row of rectangular hangars, all facing the same way.

T-hangar row

A T-hangar row is created by placing several interlocking T-hangars next to each other.

Double nested T-hangar row

The double nested T-hangar row is similar to the normal T-hangar row, but the interlocking pattern is slightly more complex.

Extensions

The extension feature is used to add a range of additions to the standard hangar, such as protruding roof on either side of the hangar, or a small walled in area.

There is no limit to how many extensions you can add to the hangar, although this is usually naturally limited by the size of the hangar. Each extension must be separated by a | (vertical bar), and each extension must start with the extension type.

Currently, there aren't any measures in place to prevent extensions overlapping with other extensions or any of the door architecture. Overlapping might cause faces to be reversed.

Extension types

There are currently six different extension types, and five of these have several optional attributes. Each attribute is separated by a comma.

The correct syntax for the extensions are as follows:

```
EXT1(size),ATTR1(size),ATTR2(size)|EXT2(size),ATTR1(size),ATTR2(size),ATTR3(size)
```

Below is a list of the current extension types, please note that all sizes are in centimetres:

D(size) – This moves the door opening either to the left or to the right (negative number to move door opening to the left, and positive number to move it to the right). This is the only extension type which doesn't have any optional attributes.

LF(size) – This will create a side extension to the left of the hangar, with the roof following the pitch of the main hangar part.

RF(size) – This will create a side extension to the right of the hangar, with the roof following the pitch of the main hangar part.

L(size)/R(size)/B(size) – These extension types create extensions to the left/right/back of the hangar, but unlike the LF/RF extensions, these allow you to manually select the height and roof pitch.

Extension attributes

Each of the five side and back extensions may contain several optional attributes, and some of the attributes may only be used in conjunction with other attributes. Some attributes may contain more than one value, separated by lowercase 'x', and more than one collection of values, separated by a semicolon.

Below is a list of the different attributes, and as with the extension types, all measurement values are in centimetres.

R – This indicates that the extension only contains a roof and posts holding the roof up. This attribute doesn't contain any values.

W – This indicates that the extension only consists of walls. This attributes doesn't contain any values.

H(size) – This sets the height of the extension at the outside edge of the extension. This is only valid for L/R/B extensions and is ignored by LF/RF extensions. If this is left out, the H(size) is assumed to be the same as the hangar height.

A(size) – This sets the height difference between the extension at the outside edge and next to the hangar wall. E.g. the height of the extension R(300),H(240),A(60), will be 240cm at the outside edge and 300cm next to the main hangar wall. If this is left out, the A(size) is assumed to be 0.

C(size1xsize2;size3xsize4) – This determines the distance from the extension's end walls to the main hangar section's end walls. A negative value will extend the extension's end walls past the main hangar section's end walls. There are several valid syntaxes for this attribute; C(size1), C(size1xsize2), C(size1xsize2;size3), C(size1;size3), C(size1;size3xsize4) and C(size1xsize2;size3xsize4) are all valid syntaxes.

For L/R/LF/RF extension types, size1 is the distance from the front hangar wall to the extension's end wall next to the main hangar wall, size2 is the distance from the back hangar wall to the extension's end wall next to the main hangar wall, size3 is the distance from the front hangar wall to the extension's end wall at the extension's edge and size4 is the distance from the back hangar wall to the extension's end wall at the extension's edge.

For the B extension type, size1 is the distance from the right-hand hangar wall to the extension's end wall next to the main hangar wall, size2 is the distance from the left-hand hangar wall to the extension's end wall next to the main hangar wall, size3 is the distance from the right-hand hangar wall to the extension's end wall at the extension's edge and size4 is the distance from the left-hand hangar wall to the extension's end wall at the extension's edge.

P(numberxsize) – This is an optional attribute for the R and W attributes. Number determines the number of posts used. For the R attribute, size is the posts' offset from the extension's edge, and for the W attribute, size is the height difference between the wall and the posts. If left out when using the R attribute, the number of posts is assumed to be 2 and the offset 0. There are two valid syntaxes; P(number) and P(numberxsize).

F(size;X) – This is an optional attribute for the R attribute only. This will create a textured floor underneath the roof. The size value determines the floor's height above the ground. X determines the floor's texture. X can either be W for wooden floor or C for concrete floor. If X is left out, the texture is assumed to be wooden floor. There are two valid syntaxes; F(size) and F(size;X).

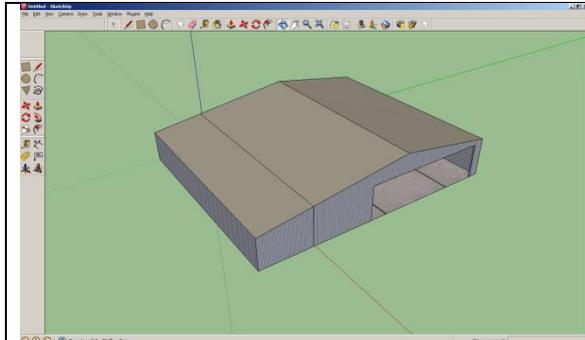
G(number) – This is an optional attribute for the W attribute used in conjunction with the P(numberxsize)/P(number) attributes. This determines which wall panels should be skipped. E.g. if the wall has eight posts, there will be seven wall panels numbered from 0 to 6, so G(1) would remove the second wall panel, whereas G(6) would remove the last wall panel. It's also possible to remove the end panels by using a negative number; G(-1) will remove the back/left panel and G(-2) will remove the front/right panel. It's also possible to remove several panels for the same extension; G(2), G(4) will remove the third and the fifth panels.

It doesn't matter which order the attributes are in, however each extension must start with the extension type; e.g. **R(300),H(230),A(40),R,P(10x15)** is the same as **R(300),P(10x15),A(40),H(230),R**

and **R(300),R,H(230),P(10x15),A(40)** but **P(10x15),R(300),R,H(230),A(40)** is invalid as P(10x15) isn't a valid extension type, but an attribute.

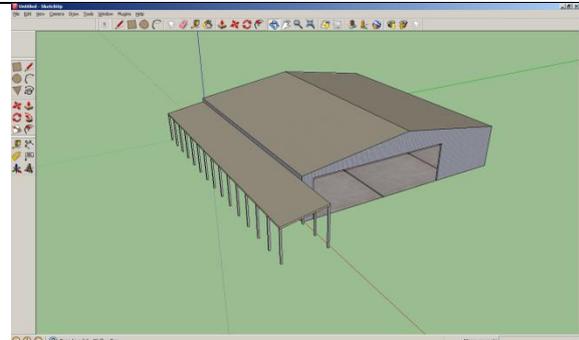
Examples

Below are several extension samples and how they would look.



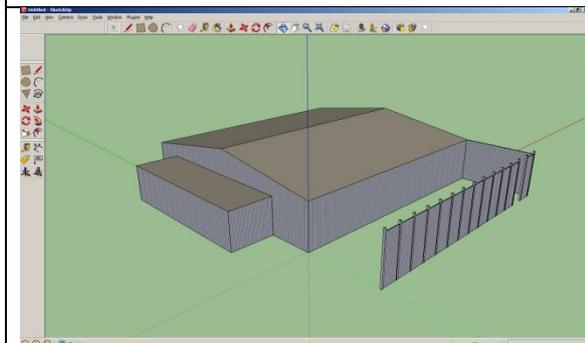
D(200) | LF(400)

The door opening has been moved two metres to the right and there is a four metre extension to the left.



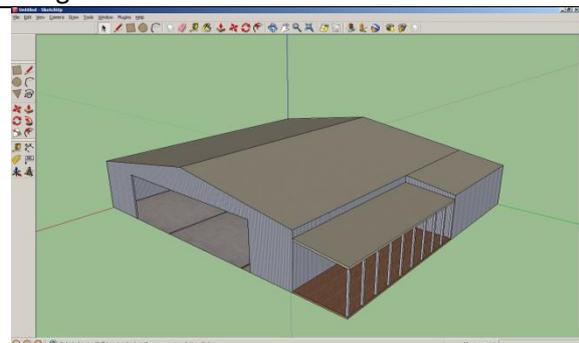
D(-200) | L(300),H(230),A(50),R,P(15),C(-300)

The door opening has been moved two metres to the left, there is a three metre roof only extension on the left, which is held up by 15 posts. The height of the extension is 2.3 metres at the edge and 2.8 metres next to the hangar wall. The extension extends three metres forward of the hangar front wall.



L(500),W,P(15x20),G(-1),G(12) | B(300),H(250),C(200x300)

This hangar is seen from the back. There is a wall on the hangar's left side. The wall height is the same as the hangar height. There are 15 posts which are 20cm taller than the wall. The back panel and the 13th panel are missing. There is a three metre extension to the back, which is 2.5 metres tall, with a flat roof, and the extension end walls are three metres from the left hangar wall and two metres from the right hangar wall.



R(300),H(240),A(30),R,P(10),C(0x700),F(10;W) | RF(300),C(956)

There is a three metre extension to the right, the first 9.56 metres is a roof only extension with 10 posts and a wooden floor, where as the last seven metres is an extension following the hangar roof's pitch.

Automated placement

Automated placement is the easiest way to accurately determine the location and size of hangars. This is done by selecting two perpendicular edges. For the standard and arched hangars, these edges represent the back and the left side of the hangar, whereas for the hangar rows and t-hangar, these edges represent the front and the left side of the hangar.

Example

I have added imagery from Ann Arbor Municipal Airport and this image shows a double nested t-hangar row, with five hangars on each side. It may be hard to see on the second screenshot, but I've drawn an edge along the front wall and the left side and selected both of them.

As you can see in hangar info box, the angle, x-pos and y-pos have been filled in automatically. After changing the front and back spaces to 5, the hangar is placed at the correct location and angle, and you can see the hangar doors match the parking markings perfectly.

