JOINT PUSH PULL (VISUAL EDITION)

jointpushpull.rb

Foreword - Installation

Joint Push Pull is a transformation that tries to offset faces by a distance along their normal while generating a contiguous shape, so a kind of hybrid transformation between classic Sketchup push-pull and Vector-oriented push-pull.

Although it was not its main purpose, the macro supports the two other forms of Push Pull: Classic and Along a Vector. Other published scripts do that in multi-face mode as well, and probably much faster[1]. The ones included in the macro provide a few extra options and manage the transfer of textures.

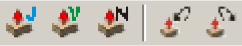
To install the script, you should unzip[2] the file you downloaded (*jointpushpull.zip* in principle) into the Sketchup Plugins folder. This operation should do the following:

- **Copy 3 script files to the Sketchup Plugins folder**:
  - ***jointpushpull.rb***, the main script
  - ***LibTraductor.rb***, a utility to manage language translation[3]. Make sure you have the latest version (at least 20 Jan 08).
  - ***LibTraductor.def*** a configuration file that can be edited to force a particular language
- **Create a sub-folder JPP_DIR** (if not already existing), where are stored icons files (***Jpp_J.png*** and ***jpp_V.png***) as well as the documentation in PDF format.

For those who might have installed the initial version of Joint Push Pull, you can delete the 2 following files, no longer useful: ***LibPlaneSelectorTool.rb*** and ***LibVectorSelectorTool.rb***

The macro will insert 5 new menu items in the menu ***'Tools'*** and create a toolbar with the 5 corresponding icons[4]:

- ***Joint Push Pull***, the main function of the macro
- ***Vector Push Pull***, provided for convenience, covering the vector-oriented push-pull in multi-face mode, with several options, axis lock and visual feedback.



- ***Normal Push Pull***, provided for convenience, covering classic Sketchup push-pull in multi-face mode (with some extra options).
- ***Start-over Push Pull***, which performs an Undo of the previous push-pull and reselect the faces initially selected (as they may have been erased).
- ***Redo same Push Pull***, which executes the most recent push pull on new selected faces, with the same options.

Compatibility: the macro works with **Sketchup v6 and should work in v5 as of v5.0.260**. I tested it on Windows XP and Windows Vista. It should normally work on Mac, based on first feedback from Mac users (but I did not test it myself).

[1] You can find some on the Crai Ruby Depot site, Smustard and in the Sketchucation Ruby forum (for instance by Pixero at http://www.sketchucation.com/forums/scf/viewtopic.php?f=153&t=1528&sid=ceeb9bb56c08b95de7aa59bcfd471794

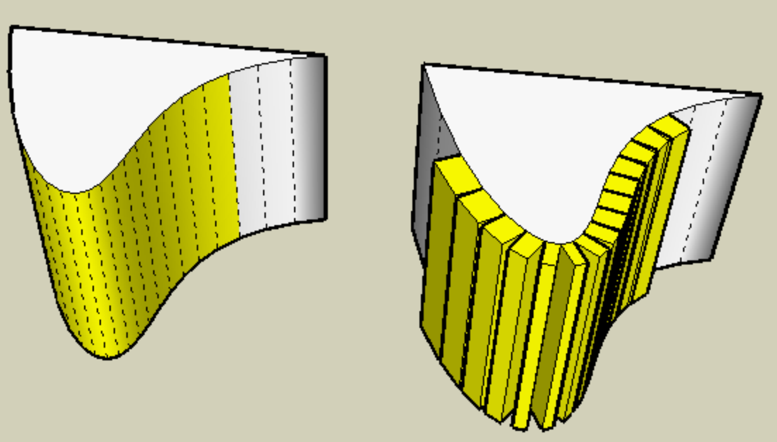
[2] In Winzip, make sure you do a Select All, and then an extract in the Plugins folder, so that you create or update the subfolder JPP_DIR.

[3] Note that this utility macro is also shared by the bezierspline script family

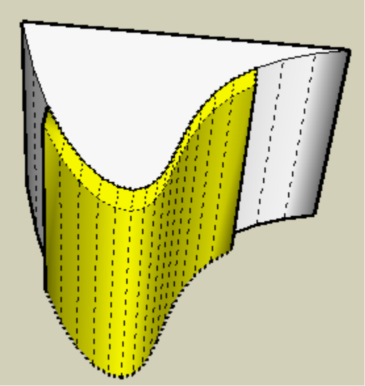
[4] Credit to **Daniel S.** (Buenos Aires), for the design of the toolbar icons

1. Overview – About Joint Push Pull

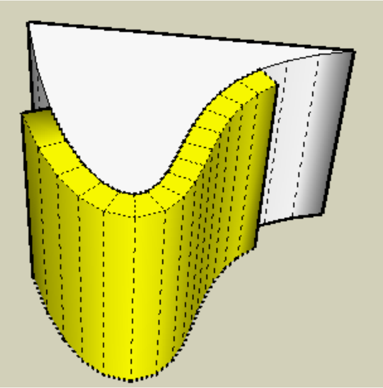
As a Sketchup user, you are familiar with the Push Pull tool, probably the most emblematic transformation in Sketchup 3D modeling. When applied to multiple faces on a curved shape, **Standard Sketchup Push Pull** does not give a nice, contiguous geometry, as shown below:



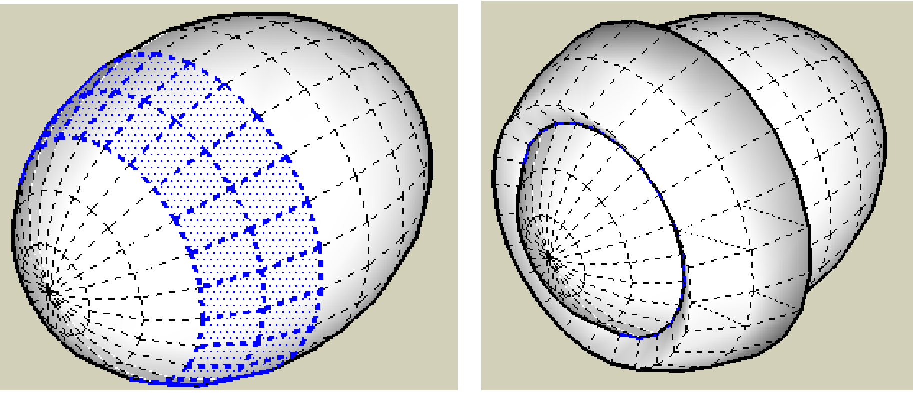
An alternative method is a **Push Pull along a vector**. This is not part of the standard Sketchup application, but there exist several plugins providing this functionality. The result is much nicer, displaying contiguous faces. However, Vector Push Pull does not generate a uniform deformation (here a uniform thickness), because geometrically, all faces are offset along the same direction, regardless of their position on the curved shape.



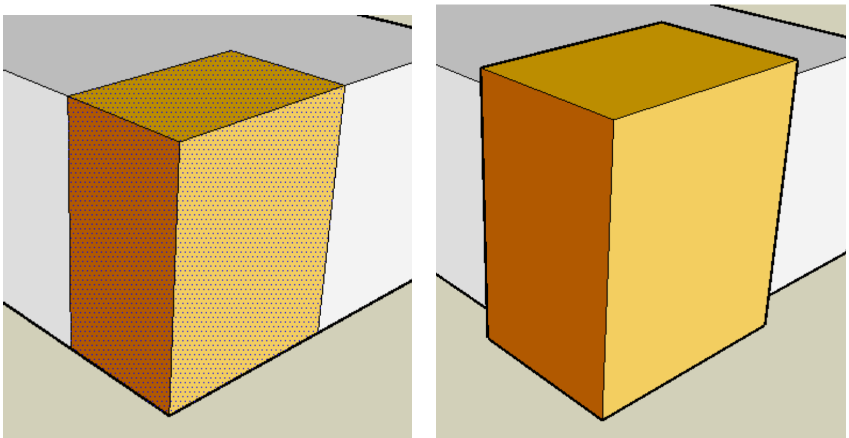
Joint Push Pull tries to make a compromise between the 2 methods: offsetting faces along their normal and generating a contiguous shape.



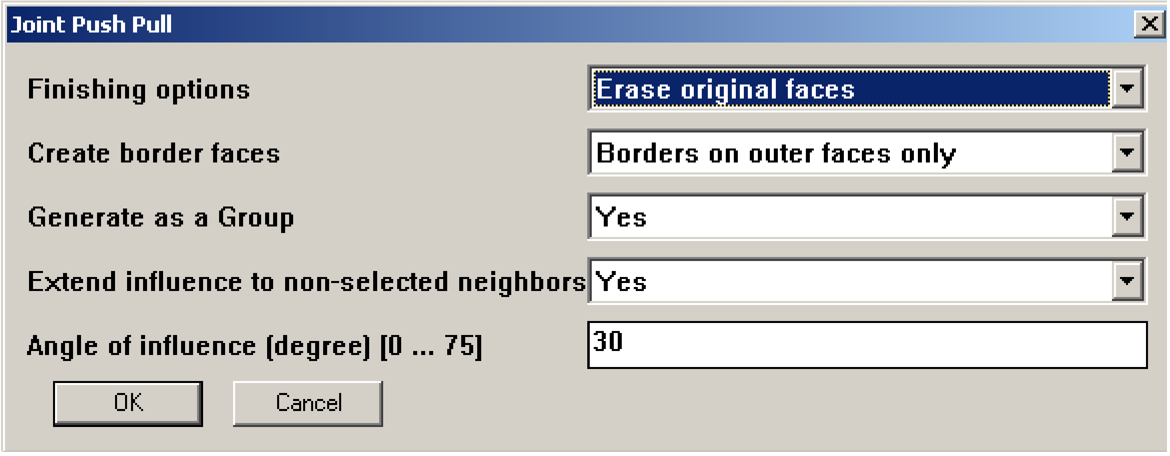
Of course there are situations where Vector Push Pull cannot make it at all. For instance, to create a ring around the balloon below:



Finally, the macro does not require that faces are located on a smooth surface. The macro works with sharp angles as well:



Because the problem is not geometrically simple and decidable in all situations, I had to leave several options for configuring the operation: the **distance of extrusion**, the **privileged plane** and **other parameters**, as shown in the dialog box below. They are explained in the rest of this tutorial:

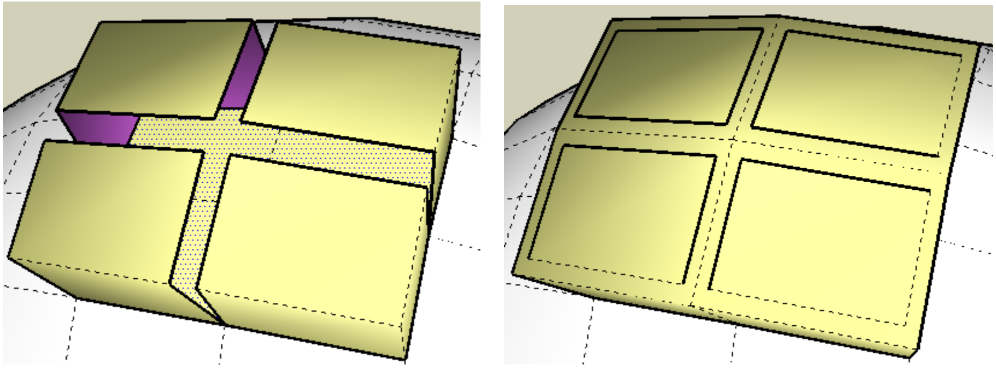


Joint Push Pull

Finishing options	Erase original faces
Create border faces	Borders on outer faces only
Generate as a Group	Yes
Extend influence to non-selected neighbors	Yes
Angle of influence [degree] [0 ... 75]	30

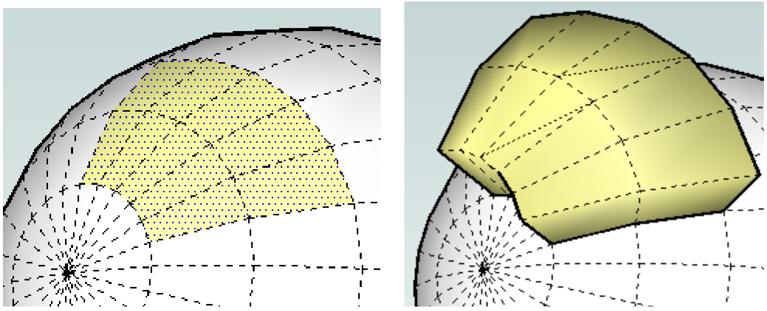
1) Principle

The principle of Joint Push Pull is to 'join' the generated faces by computing the 'best' average position of their vertices.

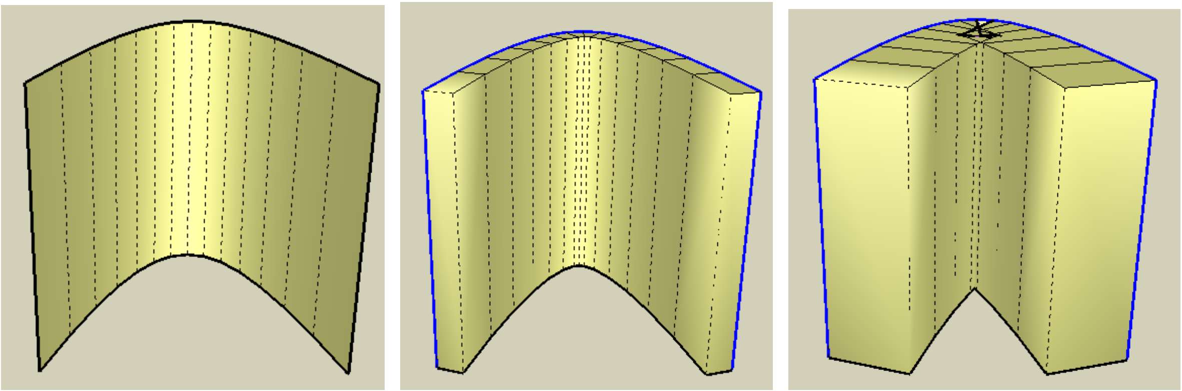


Let's make clear however that there is no exact mathematical solution in the general case. The macro uses a derived version of the Seidel-Gauss algorithm with Choleski linear matrix resolution[5].

As far as possible the macro tries to preserve the shape of the original faces, but this is not always possible geometrically because Sketchup is quite accurate on recognizing non planar configurations. So, some generated faces may have to be triangulated, even if they 'look' planar:

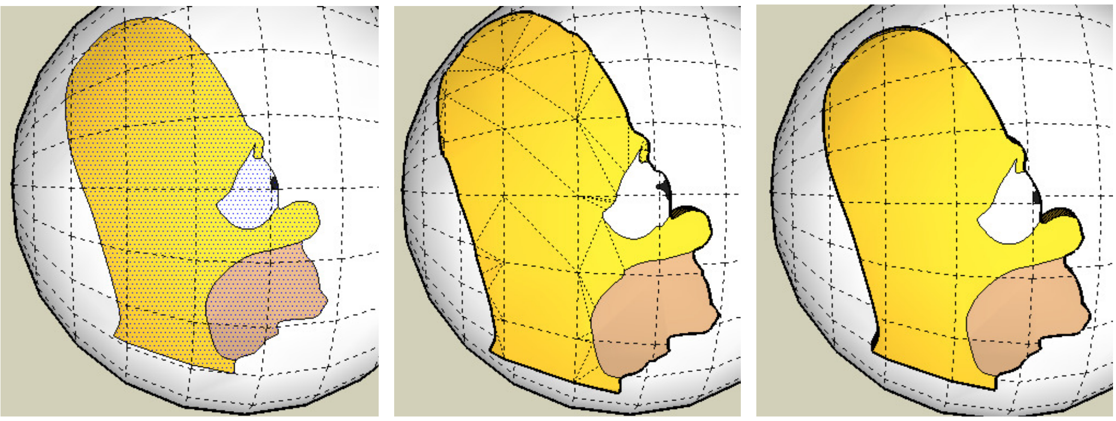


Let's also make clear that the macro does not help solving topological issues which may arise with any Push Pull transformation, in particular the overlapping of faces and borders. For instance, if you have the following shape, the result will be fine if you push-pull by a small distance, but geometrically messy if the distance is too large.

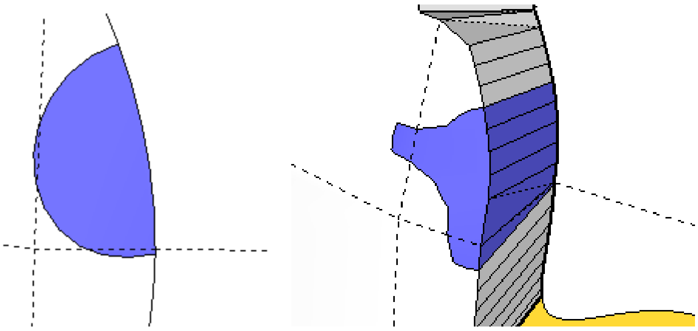


[5] This is pure joke! It just computes the arithmetic mean of normal vectors by group.

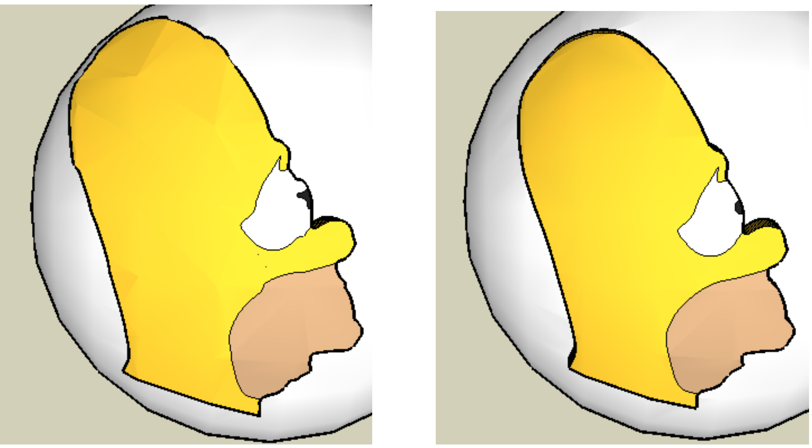
So as a general rule, you should only use Joint Push Pull, when you expect the result to be geometrically meaningful. Most of the times, the transformation will work fine when the surface is regular, but there are cases where it might even be a better compromise to use a Vector push-pull instead.



In the example above, the deformation does not give a nice result for Homer's eye. Let's zoom to understand why. As you can guess the transformation needs geometrically to enlarge or reduce the size of faces to accommodate a contiguous shape. What happens to Homer's eye is that it is spread out on several faces, each with their own deformation and dilatation along their individual normal:



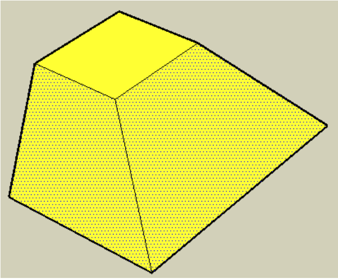
Smoothing may also become an issue in some situations. Below are the same Homer's head pushed-pulled with the two methods:



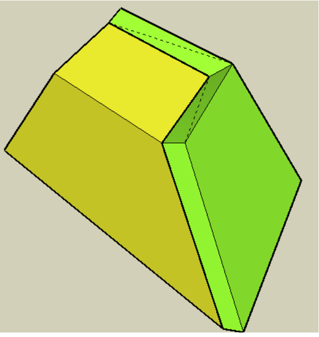
You can notice that smoothing is not nice at the top of the head (and that contours are not as soft as in the original model!). This is due to triangulation and there's little you can do, unless you edit the faces manually with the *Flip Edge* tool of the Sandbox.

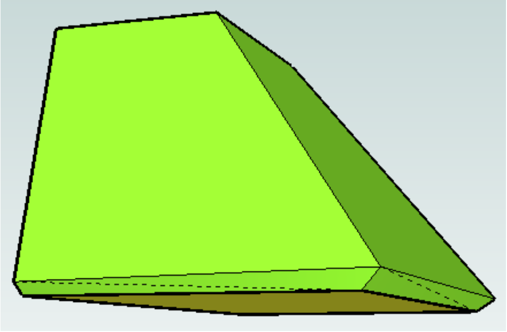
Finally, it is worth mentioning what the macro does NOT do, which is to create new faces to arrange nicely the extrusion.

Consider for instance the prism below, and let's assume we want to extrude the 2 selected faces:



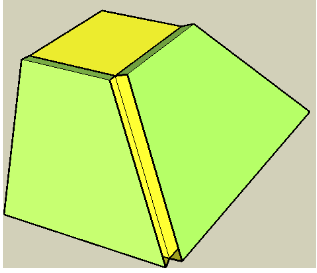
With a Joint Push Pull, here is what you will likely obtain, viewed from above and from below:

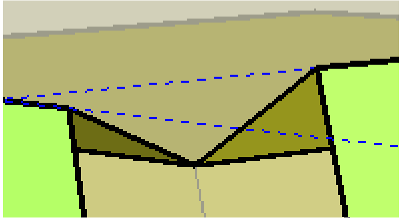




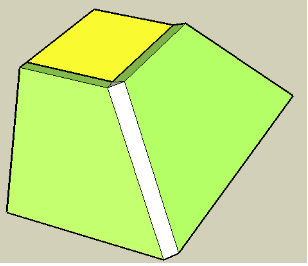
Not nice….!! However, the macro did its job. So why is it that the result is not what you would expect intuitively?

If we do a regular Push Pull on the 2 faces, we obtain a gap between the generated faces, but unfortunately, they cannot 'naturally' be joined, because the edge would not intersect:





So, in the 'real life of modeling', we would rather manually create a **NEW face** to join the borders.

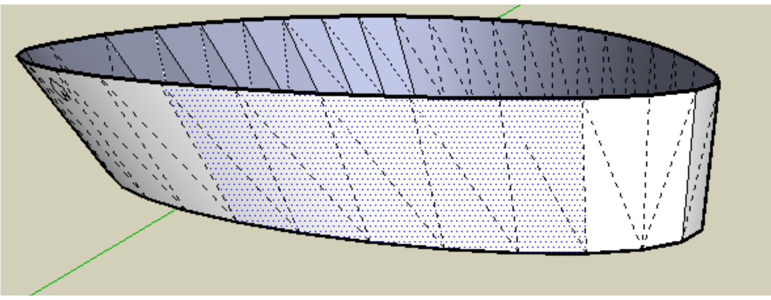


The lesson to take is that Joint Push Pull does not make it in all situations, and that you should always check if other methods would not give a better result geometrically. It also highlights that Joint Push Pull is not a scaling tool, but just a multi-face extruder.

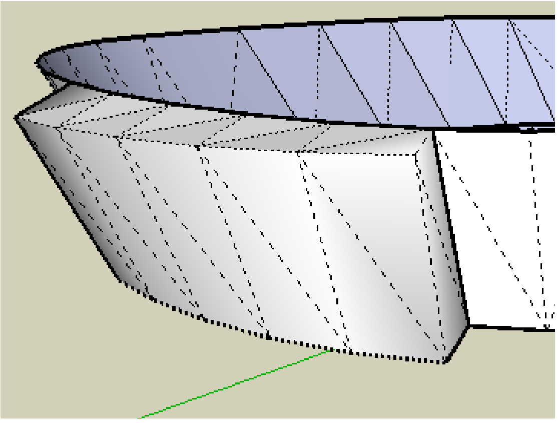
2) Privileged plane

Because the macro tries to extrude along the face normal, you may sometime obtain results that you don't really expect, even if the macro did its work properly. This is because, when you extrude with a large distance, the slight difference of angles between face normals is amplified.

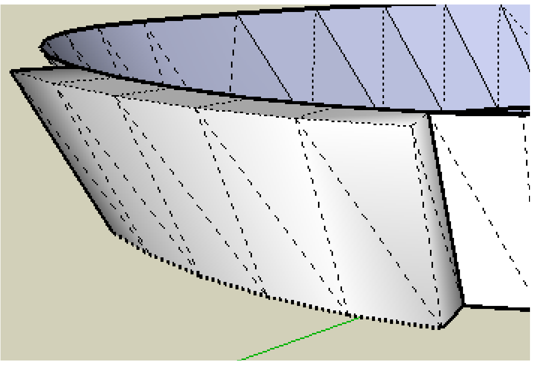
In the following example, the triangle faces look roughly coplanar but are not vertical.



However, if you extrude them, the push-pull will preserve the angles between faces and amplify these angles:



You can correct this effect by specifying a plane, here the Horizontal plane, to guide the normals, so that you will finally extrude in the horizontal plane, while keeping the curved shape:



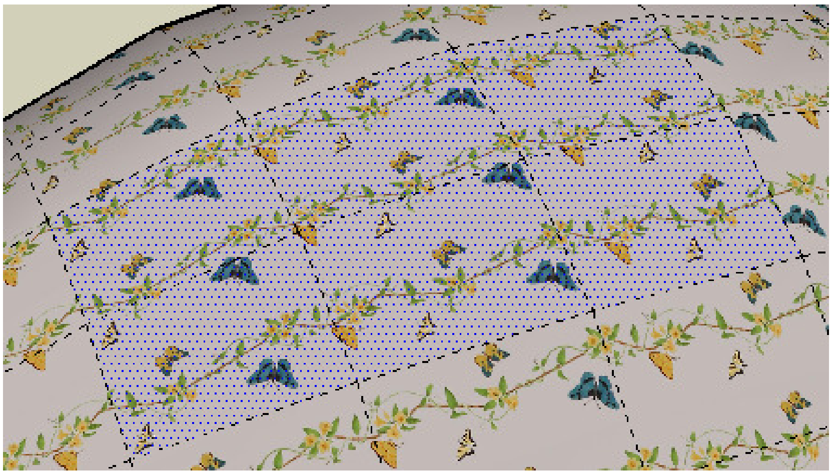
The macro allows you specifying the plane perpendicular to each 3 axis, as well as to enter graphically your own custom plane.

This technique is often a way to obtain nicer results when you have a privileged direction in your model.

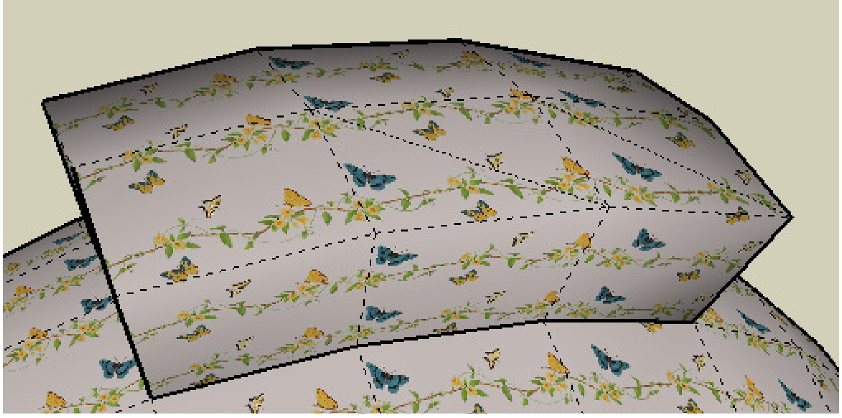
3) Textures

The macro transfers textures from the original faces to the generated faces. Furthermore it tries to accommodate the deformation of faces to provide texture continuity in the extruded model, even if triangulated. It does the same with the borders, although this is trickier. As a result, it may sometimes be a better idea to texture your model before deformation if you intend to keep continuous textures and don't thant to go the painful job of positioning textures afterward.

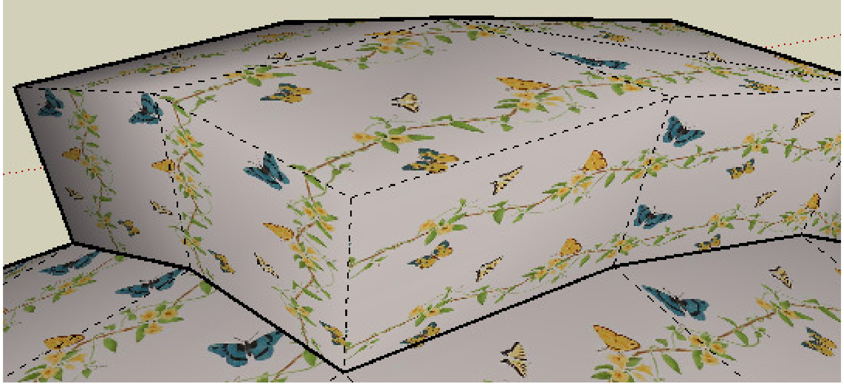
For instance, in the shape below, textures have been mapped by projection on the sphere.



The extruded shape reproduces the same image texture on the generated faces (though a little big larger), and tries to paint the border to give a feeling of texture continuity.

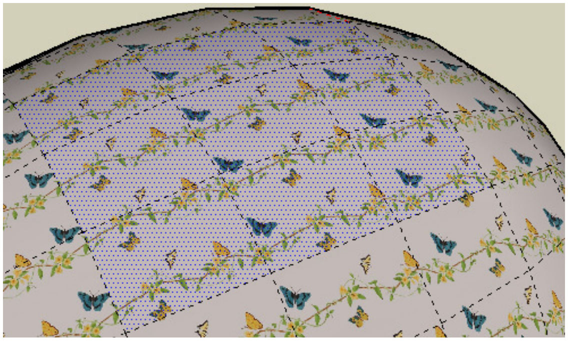


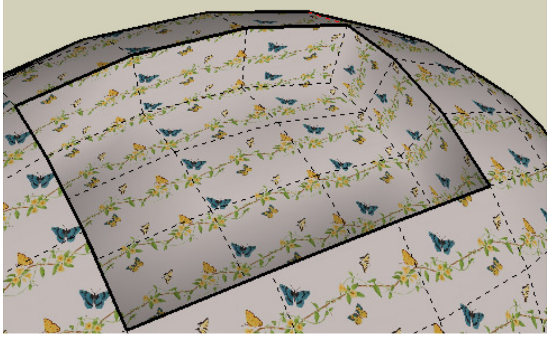
On the border however, there is no miracle, because the repetition of the texture pattern depends on the distance. So you may obtain visual texture deformation. Also, there is no geometrical solution to obtain texture continuity laterally on the borders



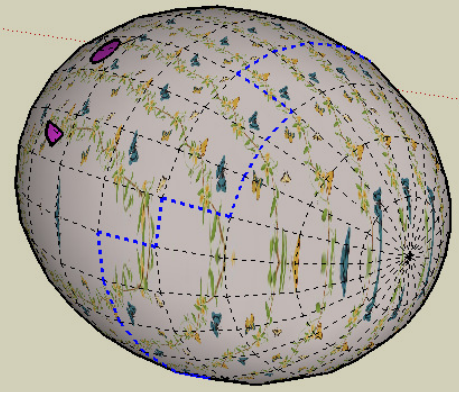
4) Carving

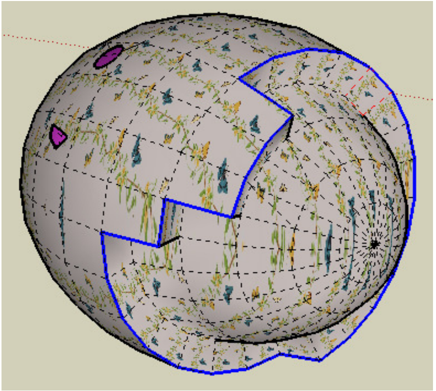
You can carve faces by entering a negative distance. The macro normally takes care of orienting the faces accordingly.





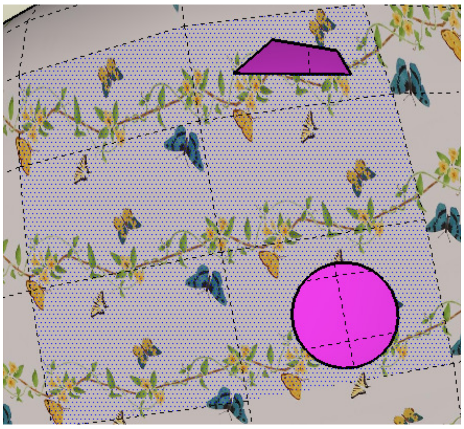
You can also shave surfaces this way

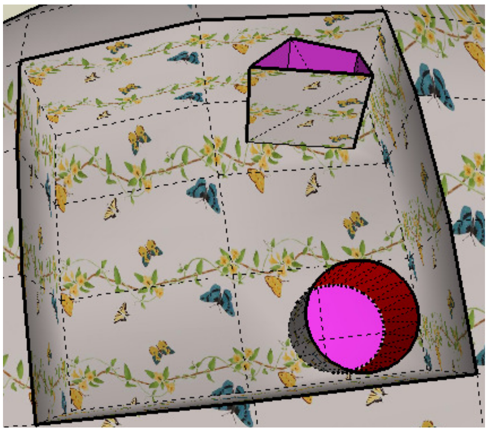




Note however that unlike the native Sketchup Push Pull, the macro does not create through-holes, when you dig too far because it has actually no clue of the topology of the model. Anyway, there is no exact mathematical algorithm in multi-face mode. The best I can suggest is to do the Push Pull with a large enough distance, then an Intersect with Model, and a clean up of the unwanted portions of the model.

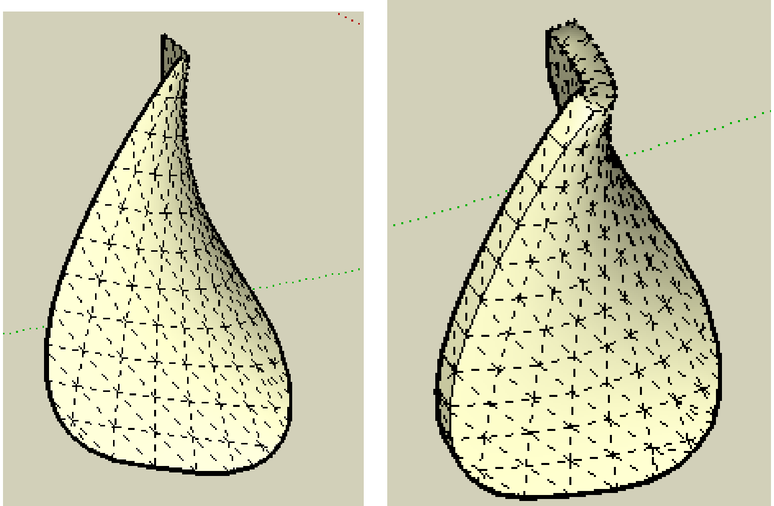
Note that holes are not carved out and generate borders, as you can expect.





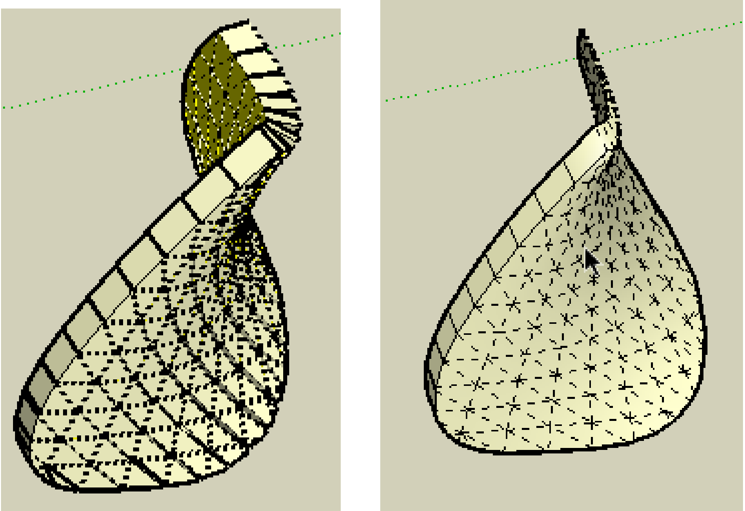
5) Thickening surfaces

A natural usage of the macro would be to thicken a curved surface. The macro provides additional support for this operation, with an option to reverse either the original or the generated faces; otherwise, you would always have one that is not correctly oriented.



The only issue is that in Sketchup, when you select a face, you actually select both sides, and, in case where both sides are textured, there is no easy way to know what is the orientation of the normal. So you may have to try with a positive or a negative distance to make sure you don't end up with back-faces on each side.

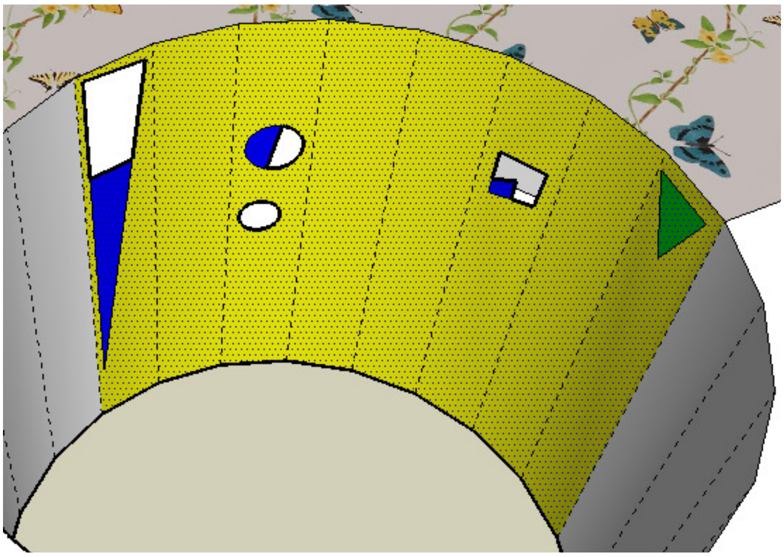
Note that the example above is a case where it would be difficult to obtain a satisfactory result with standard push pull or vector push pull, as the surface spreads around in all directions.



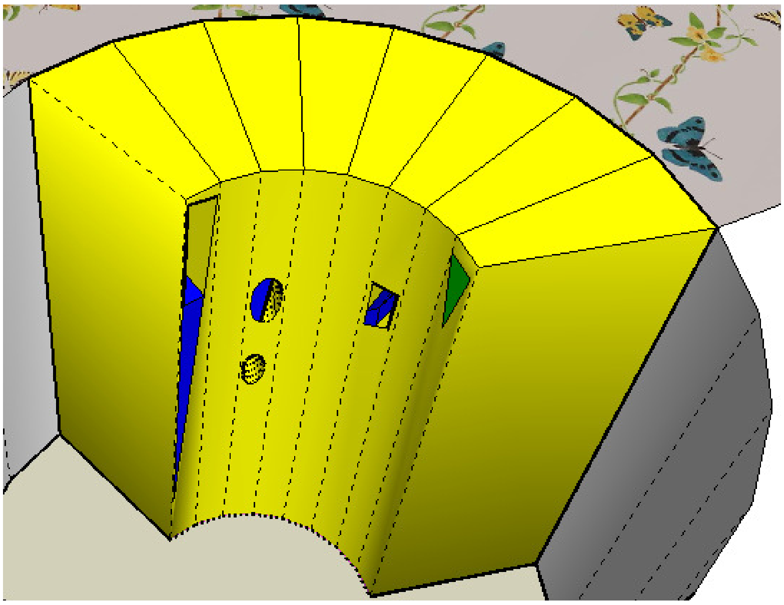
6) Holes and embedded faces

The macro supports *partially* situations where faces have holes and embedded faces (total and partial). This deserves a special treatment because you usually want holes to remain within the generated face despite its deformation. So holes have to be deformed as well.

In the example below, there are one true hole, one fully embedded face (green triangle) and 3 semi-embedded faces (in blue).



The transformation will indeed squeeze the faces to fit them in a cylinder with a smaller radius. This means that the holes and embedded faces will be shrunk accordingly, and holes will not end up as cylinders, but as portions of cones.



So depending on what you want to achieve, it may be good or bad decision to design the holes before the transformation.

Where there is a limitation currently is when holes are on a face which happens to be triangulated when transformed. This is not predictable, but if this happens, the advice is to enclose the hole within a triangle, because triangles are always planar and of course never triangulated.

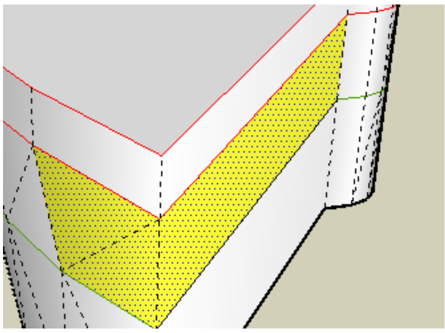
7) Influence of Neighbor faces

As you have seen, faces influence each other within the selection, in order to keep contiguity when push-pulled. This raises 2 questions however:

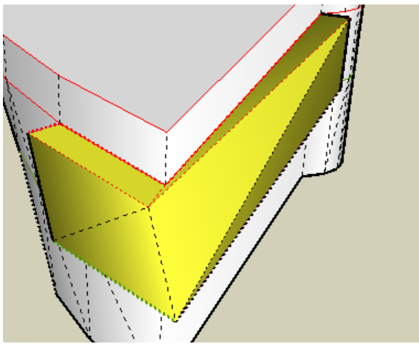
1. To what extent can we control this influence?
2. What about the faces that at the border of the selection; would they be influenced by their neighbors, even if not included in the selection?

These questions are answered by a parameter called ***Angle of Influence***. It indicates the maximum angle between the normals of 2 faces that would make them influence each others.

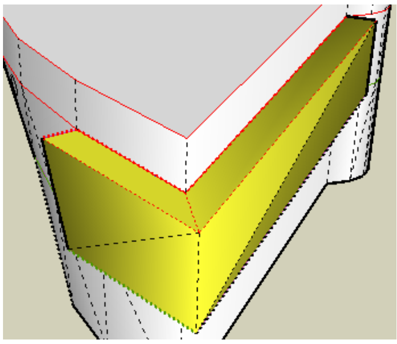
Consider the following situation, with the 3 yellow faces selected:



If you compute the average of vectors on the top vertex at the corner, the 2 triangular faces would count twice as the rectangular face. You would thus obtain a transformation where the resulting offset of this vertex would be more influenced by the 2 triangles than by the rectangle:

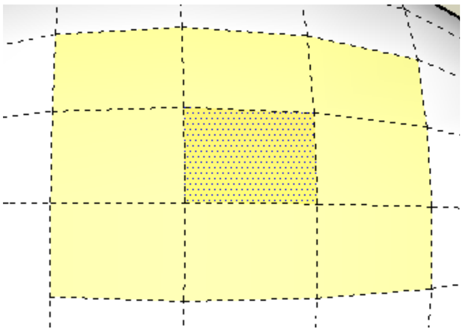


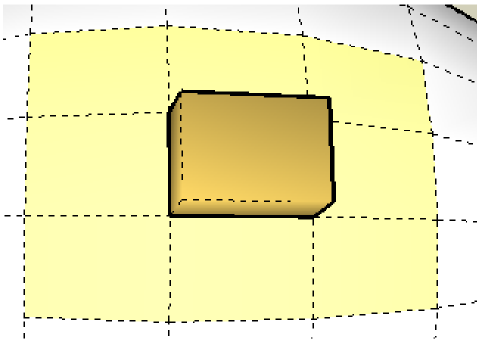
By setting the *Angle of Influence* to a value which is greater than the angle of normals of the 2 triangles, say 30 degrees to be safe, the average will be done by group, the 2 triangles on one hand and the rectangle on the other hand. So you will obtain a more natural result:



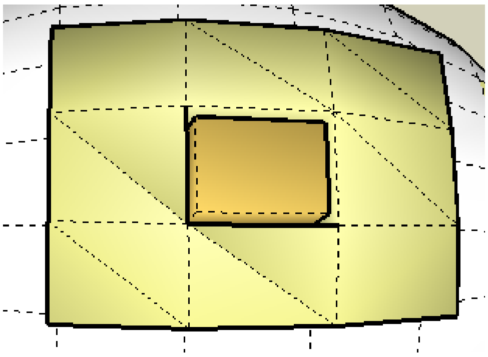
This may look technical, but this is actually the main issue that I had to resolve to make the behavior acceptable in real cases. For convenience, the macro allows setting the value of the *Angle of Influence*, but I would advise to leave it to its default value of 30 degrees.

For neighboring faces that are outside the selection, this is a matter of user choice (this is why there is an option in the macro). Let's take a simple example to explain the problem. On the sphere below, assume we extrude the selected face as if it were alone in the model (i.e. not influenced by its neighbors). We obtain a result which is actually strictly equivalent to the standard Push Pull:

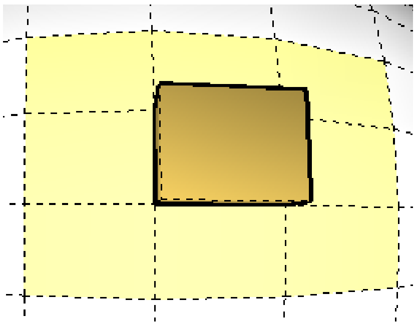


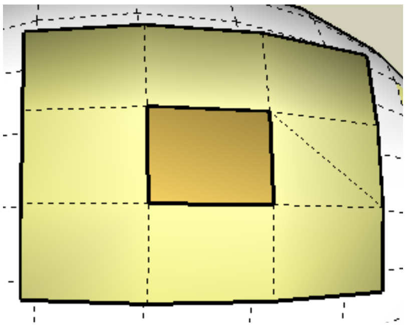


Now, if we extrude the eight faces around, we do not obtain a joint with the central face:



The option of extending influence of outside neighbors is there precisely to avoid this type of situation, whether you extrude in several rounds or you want the borders of the generated selection to reflect the general shape, and not be strictly normal:

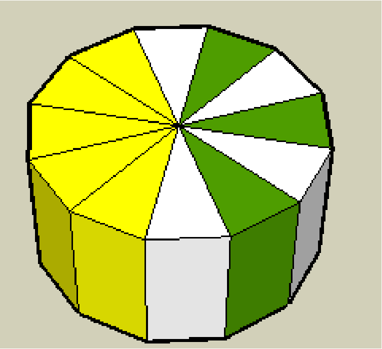




You will also notice that on regular shapes, this usually gives less triangulation of the generated faces.

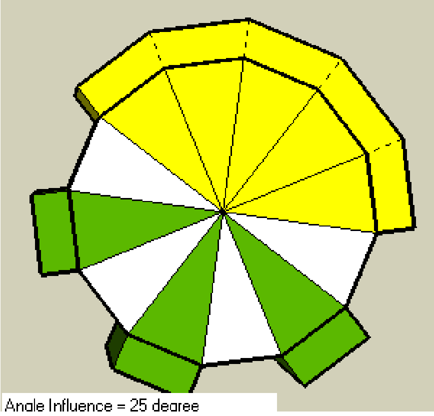
CAUTION: when you extrude in several passes, be aware that if you have the option *"Erase Original Faces"* on, you cannot do what is shown in the above example, because precisely the original faces would have disappeared in the previous push pull operation.

To make this concept of Angle of Influence a little bit clearer, let's consider the following example of a regular polygon with 12 edges[6]:

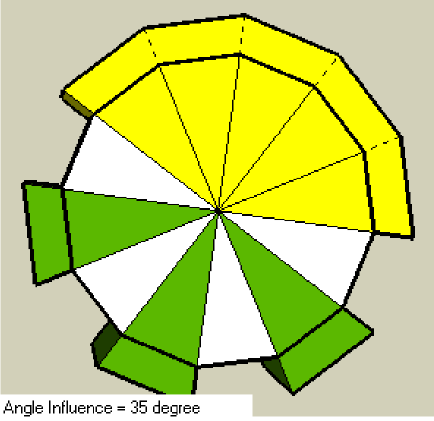


The angle between the normals of two adjacent faces is exactly 30 degree.

If we extrude the colored faces with an Angle of Influence of **25 degree** (so < 30), the non-selected adjacent faces won't influence their selected neighbors, so that the external borders will be exactly perpendicular to the edge of the polygon.



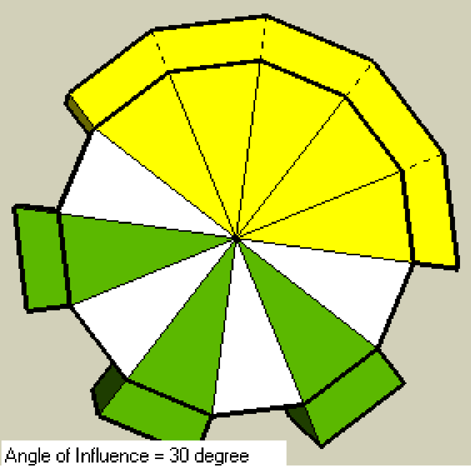
But if we increase the Angle of Influence to a value over 30 degree (say **35 degree**), we will obtain borders following the radius.



Because you may want one or the other result, I left you the choice of the parameter in the dialog box.

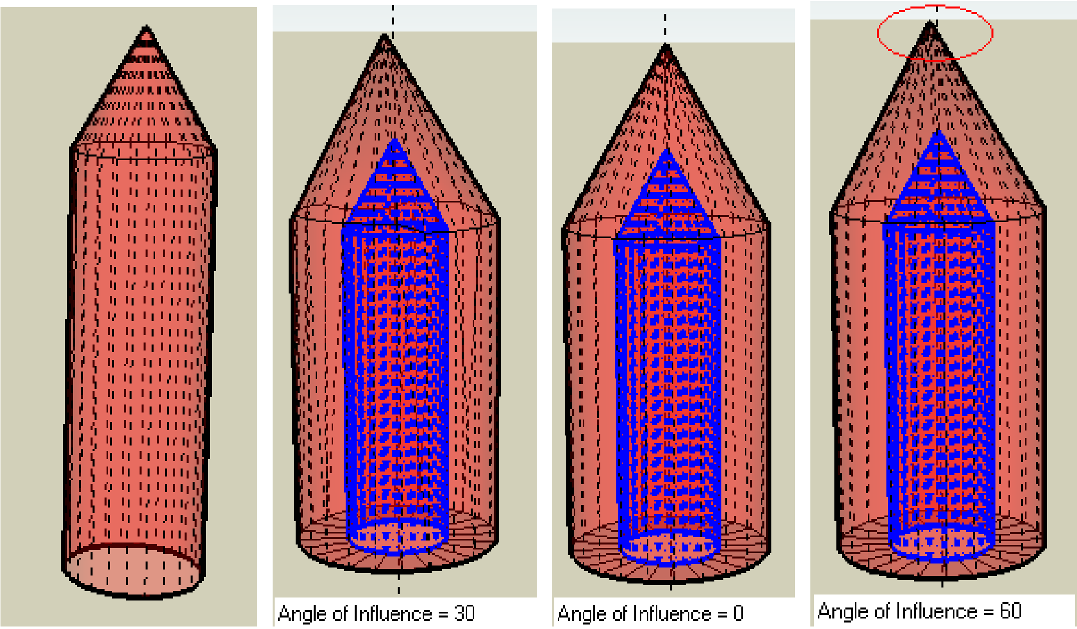
[6] The example comes from a discussion on the Sketchucation forum initiated by GreyHead.

Where it becomes weird is when you enter an Angle of Influence of exactly **30 degree**. Because of the rounding up in the calculations, you will obtain an unpredictable mix of both types of borders.



Angle of Influence = 30 degree

Finally, I want to leave to your thoughts another case where playing with the Angle of Influence can help you to obtain different results (and hopefully the one you're looking for!). Consider how this sharp pen is transformed with different value of the Angle of Influence:



Angle of Influence = 30

Angle of Influence = 0

Angle of Influence = 60

It looks like the most appropriate value is 0 degree. With 30 degree, the contours of the cone are not regular. With 60 degree, the generated envelope is not exactly co-axial with the original cone. Why? Well…. I don't now too much, but this surely means that the Angle of Influence may have much more importance than I thought originally! The fact is that the faces on the cylinder are influenced both by their neighbors on the cylinder but also by the adjacent faces on the cone (each vertex shares 4 faces).

8) Other options

The macro lets you specify a few other options on how the push pull operation is rendered:

1) Finishing Options

There are 3 finishing options, depending on what you want to do:

- **Erase original faces:** This is normally the natural option, as in Classic Push Pull
- **Reverse faces (for thickening)**: When you thicken a surface, it is of course a better idea to preserve the original face. The macro will reverse either the original or the generated face (depending on whether the distance is positive or negative) to have the textures on the external sides of the shape.
- **Keep original faces:** This can be associated to the "Generate in Group" option, if you prefer to have a non destructive approach.

2) Generate border faces

By default, only the borders on outer faces are generated. You can however change this option and specify 'No Border" when you just want to offset faces.

3) Generate as a Group: YES / NO

You have the option to create all generated entities (faces and borders) in a group, again to have a careful, non destructive approach. Be aware however that there is only ONE group created for ALL faces generated.

2. Joint Push Pull – User Interface

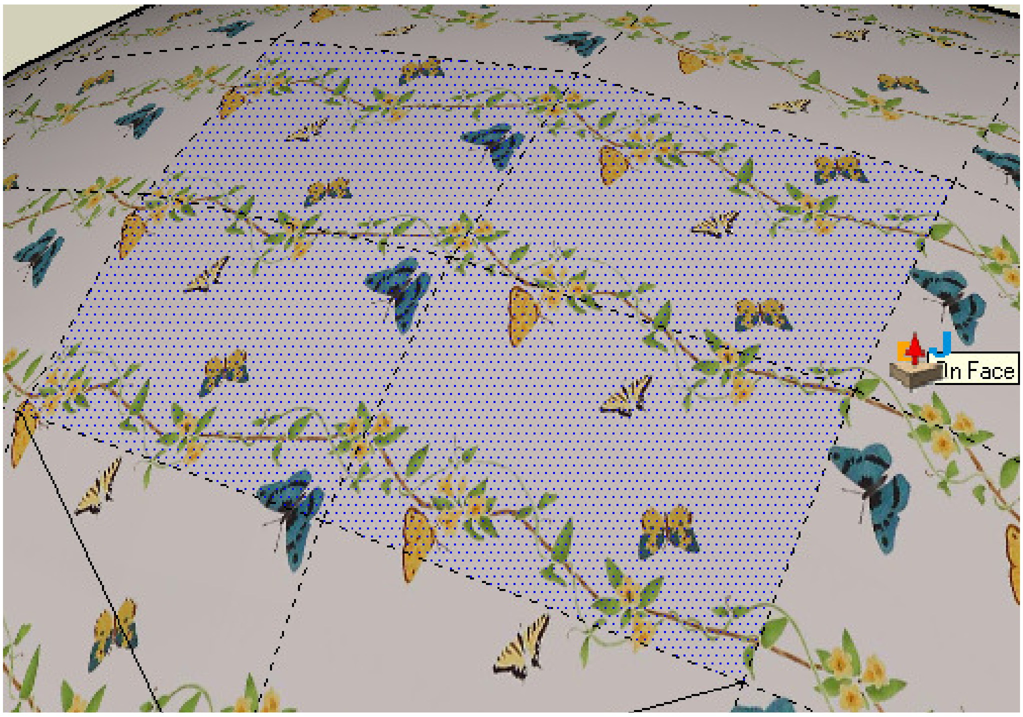
This section describes the 'visual' user interface to Joint Push Pull. The whole idea was to let you specify the distance of extrusion graphically, with inference, and get an immediate feedback of the result, unfortunately limited to a wireframe mode.

To enter in the Joint Push Pull Editor, you must first select at least one face in the model. The face selection has to be explicit, that is you cannot just select a group or a component instance to mean that you want to select all its faces.

However, faces do not need to be contiguous and can therefore be in different places in your model (if this is meaningful to you to apply the push pull with the same parameters to all faces at once).

Likewise, the macro will only consider faces in the selection. So if you select other entities, like edges, construction line and points, images, groups, component, etc…, they will simply be ignored.

Then, by invoking the Joint Push Pull operations, either via the menu item in ***"Tools"*** or the toolbar button, you should enter in the graphical input mode of Joint Push Pull. You will notice that the cursor shape is changed to the corresponding icon.



1) Principle of the user interface

The Editor has 3 states, which are in order of sequence:

1) **Face Selection**: in this state, you can move the mouse cursor over the model and select a face to be the reference for dragging all faces.

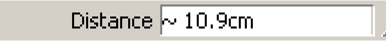
2) **Pushing / pulling faces**: in this state, you can freely drag the faces and get visual feedback in wireframe by moving the mouse cursor.

3) **Ready for Execution**: Once you finish dragging faces, you are ready to start the Joint Push Pull operation by ***Double-Clicking***, pressing ***Enter*** or selecting *"done"* in the contextual menu. Optionally, you can modify some parameters and see their effect with visual feedback (this can be useful for Privileged plane, Angle of influence in particular).

Note that you can navigate back through these states by pressing the ***Escape key***.

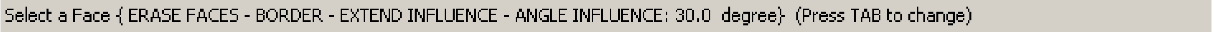
In any of the above states, you can change all the parameters of the transformation. All settings are preserved from one push pull operation to the next one. These settings are indicated in three places:

- **The distance of extrusion** can be typed and is shown in the **VCB**



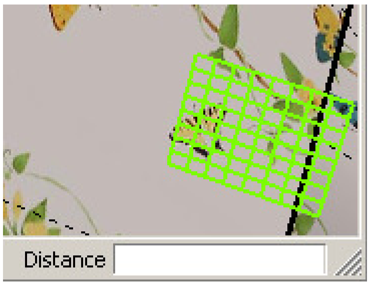
Distance ~ 10.9cm

- **The status bar** shows the current value of the five other parameters: **Finishing** options (Erase original faces, Keep original faces, Thickening), **Border** option, **Group** generation, **Extended Influence** from outer faces and **Angle of Influence**.



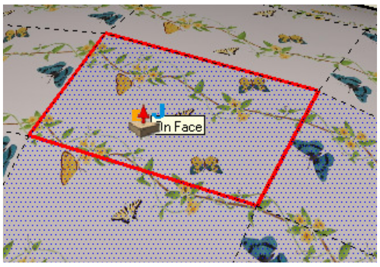
Select a Face { ERASE FACES - BORDER - EXTEND INFLUENCE - ANGLE INFLUENCE: 30.0 degree} (Press TAB to change)

- Finally, if you have specified a **Privileged Plane**, it will be displayed in a **small vignette in the lower right corner of the view**:

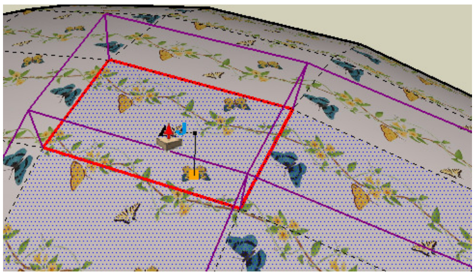


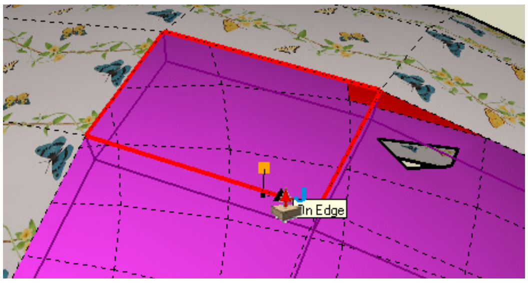
2) Selecting a face and dragging the selection for visual feedback

Move the cursor over the selected faces. You notice that each face is highlighted with a Red frame.



If you then click with the left mouse button, you can drag the face in either direction, positive or negative distance, and see the result as a purple wireframe.





The macro supports two entry modes:

- Click down on left mouse button, drag and release
- Click on left button and release, then move the cursor to drag faces and click again when done.

The distance is indicated in the VCB in the current units used in your model. Note that the distance is shown as a positive or negative number depending on which direction you drag the faces.

3) Using Sketchup Inference

The editor makes full usage of the Sketchup inference mechanism so that you can visually adjust the distance by reference to a particular point in your model, as in most Sketchup tools. However, when you drag, you may want to skip some inferences. For this, just keep the ***SHIFT key pressed down***, while dragging faces.

4) Entering the distance manually

At any time you can type a distance in the VCB. There are a few subtle hints to keep in mind:

- If you did not select & drag yet, then the Joint Push Pull operation will start right away after you typed Return. Positive distance means "Extrude", negative distance means "Carve". This is a kind of 'fast mode'.
- If you already selected a face and started dragging, then a positive distance will be interpreted as 'going' in the same direction as your current move. However, a negative distance will be interpreted in absolute term and will force the 'Carving' direction. In any case, you will stay in the Editor mode, so that you can preview the result. To launch the Joint Push Pull operation, you need to press Enter again or Double click.

5) Specifying a privileged plane

The default behavior of the macro is to extrude the faces along their individual normal. However, as seen earlier, by imposing a privileged plane, you may obtain nicer results in some situations where the overall shape is perpendicular to a particular plane.

The current privileged plane is indicated in the lower right corner of the view window. If there is nothing displayed, then you do not have a privileged plane.

The options are the following:

- **N: No plane – Direction = normal to face**

  There is simply no privileged plane, which corresponds to usual situations when the overall shape spreads in several directions.

  Shortcut: CTRL + ARROW DOWN

- **Axis Planes (B, R, G)**

  For convenience, the macro allows to specify directly for the axis planes, as in many cases your model will be oriented along the axes. The plan eis specified by the color of the axis which it is perpendicular to. Here are the short cuts

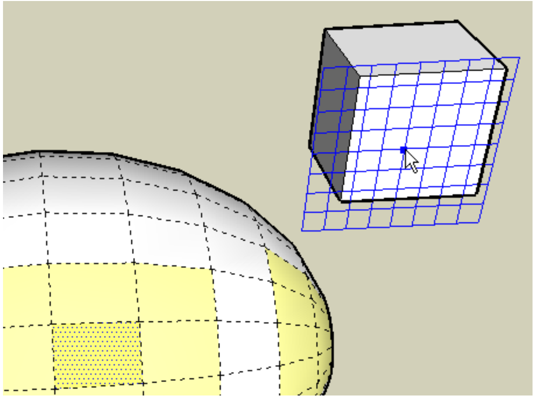
– **Blue plane** (i.e. horizontal):	CTRL + ARROW UP
– **Red plane** (vertical):	CTRL + ARROW RIGHT
– **Green Plane** (vertical):	CTRL + ARROW LEFT

- **Custom Plane**

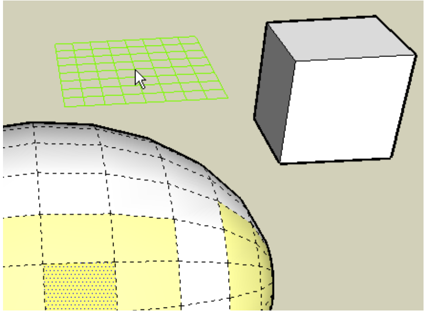
  This option allows you entering graphically a plane in the model itself. The short cut is CTRL ALONE. This will bring you to another graphical tool, where the plane will be specified with the help of a plane grid:

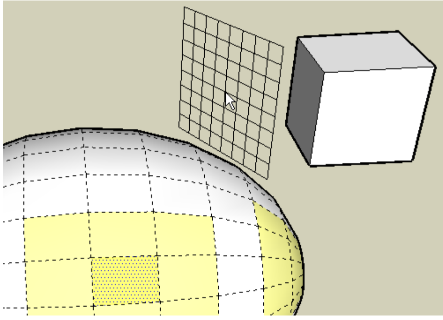
  To enter a new plane within the model after closing the dialog box with OK, you just have to position the **Plane Grid** as follows:

  - **If dragged on a face**, it will take the plane of the face. By keeping the Shift key pressed, you can preserve the direction and move it around (like with the Protractor tool).



- **Outside any face**, the default direction can be oriented using the UP, RIGHT and LEFT arrows to make it perpendicular to the axes (another way to specify n axis plane!). You can also type an angle in VCB (in degree) to orient the face.



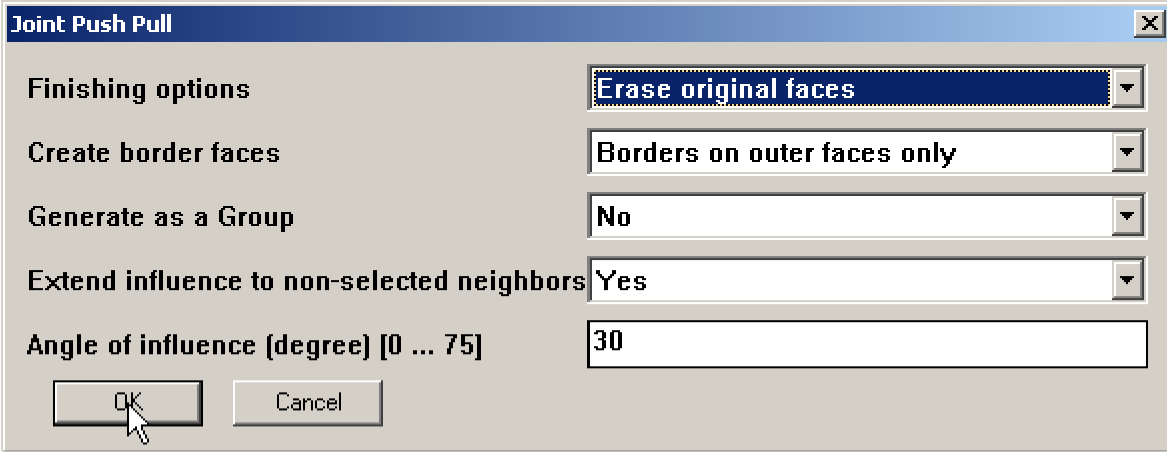


The default plane is preserved when you move the mouse around, so that you can always retrieve it by moving away from any face.

Note that only the direction of the plane does matter. The position of the origin of grid has no role.

6) Specifying all other Parameters

All other parameters can be specified via a dialog box, which you can call via the shortcut **TAB**.

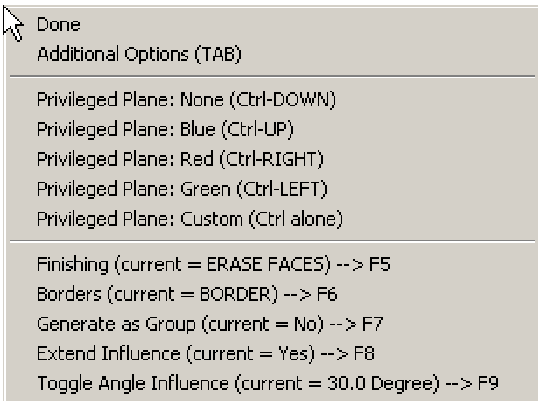


Joint Push Pull

Finishing options	Erase original faces
Create border faces	Borders on outer faces only
Generate as a Group	No
Extend influence to non-selected neighbors	Yes
Angle of influence (degree) [0 ... 75]	30

There are however other ways to specify individually these parameters by using keyboard shortcuts as described in the following sections.

Note that the contextual menu (Right Click) also gives access to the parameters.



Done
Additional Options (TAB)

Privileged Plane: None (Ctrl-DOWN)
Privileged Plane: Blue (Ctrl-UP)
Privileged Plane: Red (Ctrl-RIGHT)
Privileged Plane: Green (Ctrl-LEFT)
Privileged Plane: Custom (Ctrl alone)

Finishing (current = ERASE FACES) --> F5
Borders (current = BORDER) --> F6
Generate as Group (current = No) --> F7
Extend Influence (current = Yes) --> F8
Toggle Angle Influence (current = 30.0 Degree) --> F9

7) Finishing Options

There are 3 finishing options, depending on what you want to do:

- **Erase original faces:** This is normally the natural option, as in Classic Push Pull

- **Reverse faces (for thickening)**: When you thicken a surface, it is of course a better idea to preserve the original face. The macro will reverse either the original or the generated face (depending on whether the distance is positive or negative) to have the textures on the external sides of the shape.

- **Keep original faces:** This can be associated to the "Generate in Group" option, if you prefer to have a non destructive approach.

The shortcut **F5** allows you to toggle between these 3 options.

Note that if you select *'Keep original faces'* or *'Reverse faces'*, the faces will not 'disappear' while you drag with a negative distance. This is to give you the feedback of what we you would get once you proceed with the operation.

8) Generate border faces

By default, only the borders on outer faces are generated. You can however change this option when you just want to offset faces.

The shortcut **F6** allows you to toggle between these 2 options.

The wireframe will reflect the presence or absence of borders.

9) Generate as a Group: YES / NO

You have the option to create all generated entities (faces and borders) in a separated group, again to have a careful, non destructive approach. Be aware however that there is only ONE group created for ALL faces generated.

The shortcut **F7** allows you to toggle between these 2 options.

The wireframe will reflect the option by changing the color of the wireframe:

- **Purple** when NO generation in group
- **Orange** when generation in group

10) Extend influence to non-selected neighbors: YES / NO

We have seen the objective of this option in the previous sections. It should normally be YES, unless you want the external borders to be strictly normal to the outer faces.

The shortcut **F8** allows you to toggle between these 2 options.

The wireframe will reflect the effect of the change (note that there might be no effect in situations where you do not have outer faces, or the Angle is influence is such that the option does not have an impact on the transformation.

11) Angle of influence (in degree)

This corresponds to the maximum difference of angles between the normal of 2 adjacent faces (selected or outside neighbors) that can be averaged in the same group. In most case, leave the default value of 30 degree. Refer to previous sections for more details on how this parameter plays a role in Joint Push Pull.

If you wish to change the value, then you must invoke the dialog box by the short cut **TAB**.

For convenience, the macro provides a short cut, **F9**, to toggle between the current value and 0 degree, so that you can see the effect of having no angle of influence.

3. Classic Push Pull

Classic Push Pull is the same geometrical transformation as the standard Sketchup Push Pull, but in multi-face mode. As said in introduction, there exist already several Ruby scripts performing such operation, most based on the native Sketchup push pull transformation (whereas the macro here does not transform the face but recreates it).

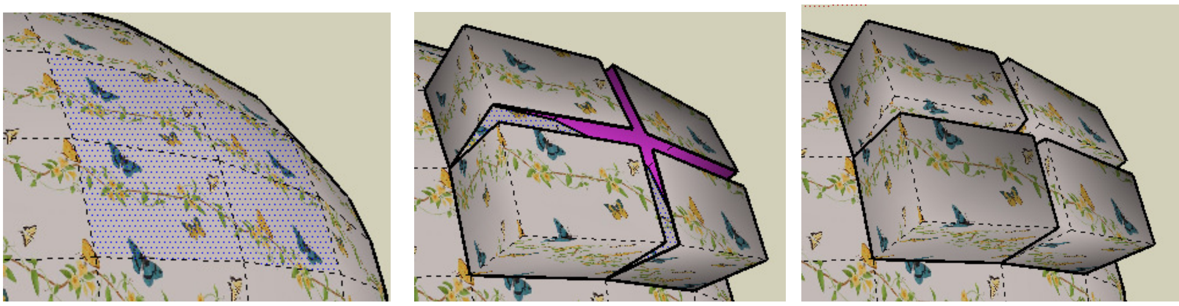
1) Overview of Classic Normal Push Pull

The Normal Push Pull version included in the present macro is actually just a derivative work from the Joint Push Pull (indeed simpler). As such, it includes the extra options and the visual feedback we have seen for Joint Push Pull:

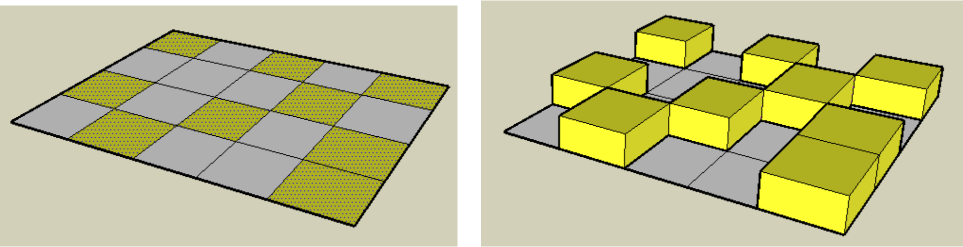
- Textures are transferred to the generated faces. Borders receives the texture of the original faces, with continuity between the original and the generated face (unlike the Sketchup Push Pull).
- Drawing attributes are also transferred (edge colors and smooth / soft state)
- Holes and embedded faces are fully supported
- The options "Generate Borders", "Create in a Group" and other finishing options (preserve original faces and thickening) are available.

There are also some differences:

- **First, the macro allows creating or not borders for interior edges**. This gives you the possibility to join generated faces the way you want.

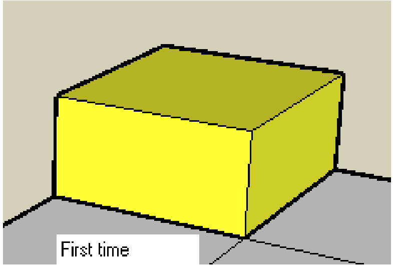


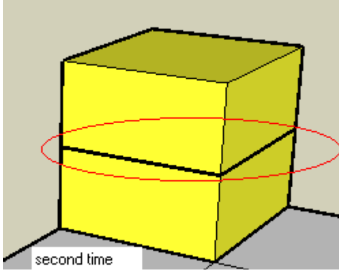
Note however that if you select faces in non contiguous parts of your model, the borders will always be generated on the exterior of each faces.

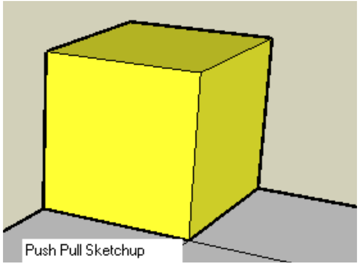


- **Second, the macro does not create through-holes when carving faces**. As said earlier, you'll have to use Intersect in Model to finish the hole.

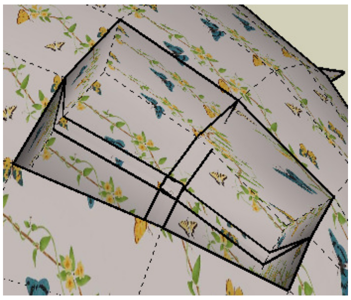
- **If you apply Push Pull a second time, the macro will not erase the coplanar border edges to make a continuous shape**. Yo'llhave to do it manually afterward.

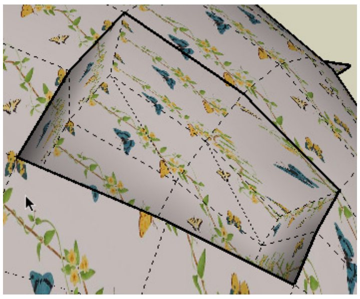






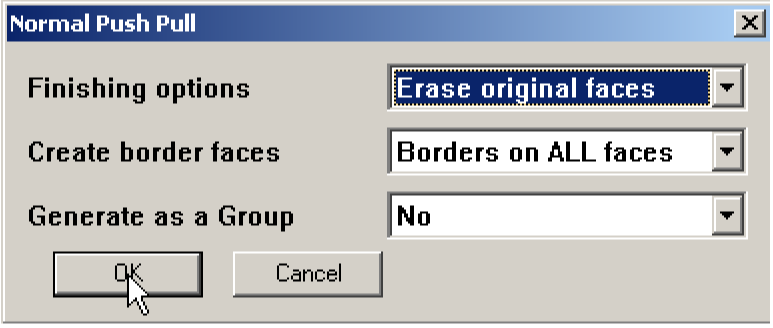
Finally, keep in mind that the Classic Push Pull, like the native Sketchup version, does not do anything to optimize the topology of the transformed elements. If you try to carve a set of faces on a sphere, this is what you will obtain (*left*), as opposed to the result with Joint Push Pull (*right*)





2) Using Classic Push Pull

The user interface for Classic Push Pull is the same as for Joint Push Pull, except that there is less parameter: Influence options and Privileged plane are not applicable. Here are the only parameters you can play with:



Normal Push Pull

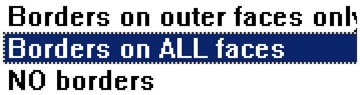
Finishing options: Erase original faces

Create border faces: Borders on ALL faces

Generate as a Group: No

For the rest, the behavior is the same as for Joint Push Pull. I suggest you refer to the section 2 above for the details on how to use the visual Editor.

There is however one subtle difference for the Borders feature. Actually you have 3 options instead of 2:

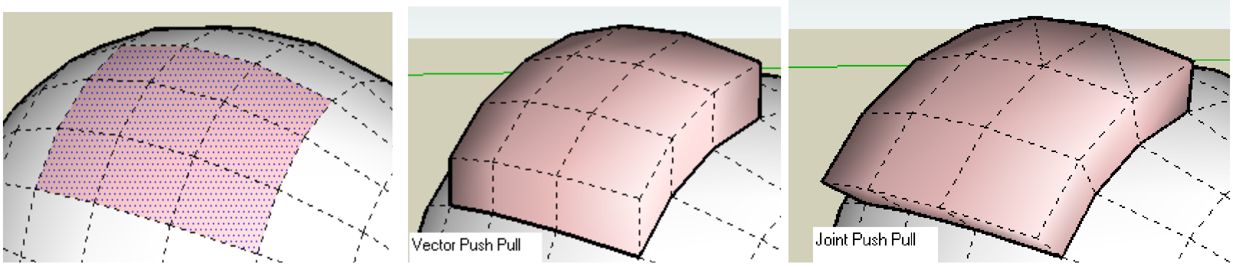


Borders on outer faces only
Borders on ALL faces
NO borders

The option ***"Border on ALL faces"*** allows you generating borders on all edges, as long as the bounding generated faces are not coplanar. This option is more in line with what you would get with a standard Sketchup Push Pull.

4. Vector Push Pull

The principle is to extrude each face along a single direction, irrespective of the orientation of the faces. So it is a simple translation by a vector, with creation of the borders. The benefit is that all generated faces will reproduce the contiguity of the original faces, with an exact continuity of the edges (as said, in some situation, it can give better result than Joint Push Pull).

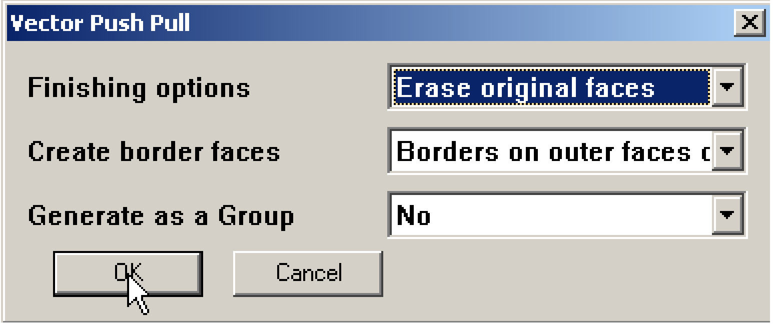


Here too, there are several existing scripts out there that do it. So don't take ***jointpushpull.rb*** as your only source for that type of transformation.

1) Parameters for Vector Push Pull

Vector Push Pull is also a Sketchup tool where you can visually enter the direction vector manually and observe the feedback in wireframe.

Apart from the distance and the direction, the other parameters are dealing with Finishing options, Borders and Group generation, which you can set by invoking the global dialog box, via TAB, or by typing the functions keys F5, F6 and F7 respectively, as seen for Joint Push Pull:



Vector Push Pull

Finishing options: Erase original faces

Create border faces: Borders on outer faces c

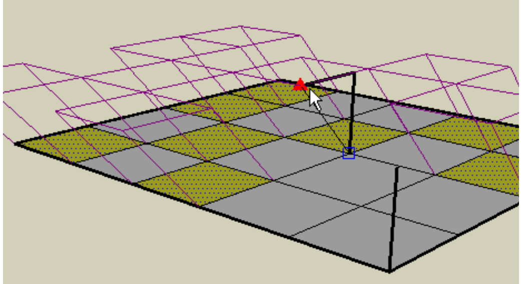
Generate as a Group: No

2) Specifying direction vector for Vector Push Pull

When you start the Vector Push Pull tool, you go to an Editing mode where you specify the vector and distance of extrusion (negative or positive) by:

- **Either clicking on an origin and an end point**; note that you will most likely use an existing segment or rely of Sketchup Inference mechanism to do so (maintain Shift down to skip inferences).
- **Choose an axis, by pressing the corresponding Arrow key**.

You then finish the input by a double-click or pressing the Enter key or entering a distance in the VCB.



5. Undo and Redo

These 2 commands are applicable to the 3 kinds of Push Pull. They act however on the last class of Push Pull used.

1) Undo and Reselect

When you executed a Push Pull operation, you may not obtain the expected result and have to try again with different parameters. Even if you perform an 'Undo' right after the operation, you may not retrieve the original faces selected (in particular if you have used the option *"Erase Original Faces"*.

As it may represent time to carefully select faces to be extruded, I found useful to provide a short cut for that.

Note that this Undo and Reselect command only work on the last Push Pull provided no other constructive or destructive operation or transformation took place in between on the model.

2) Redo last Push Pull

This command is fairly natural. It just perform Push Pull on a selection of faces with the exact same parameters used with the latest Push Pull.

Note that the command is common to the 3 categories of Push Pull: Join, Vector and Normal.

6. Ruby API to Joint Push Pull *(for developers only)*

I found useful to design an API to Push Pull operations that could be used in other scripts.

First, you need to include a statement `'Require "jointpushpull.rb"'` in the header of the calling script.

Here are the specifications of the **single API method**:

```
Status = JointPushPull.Api_call ( type="J", selection=nil, distance=0,
                                  vector=nil, finishing='K',
                                  borders='A', group=true,
                                  influence=true, angle=30.0)
```

ARGUMENTS

- `Type` is the **type of Push Pull**, as a *single character string*,
  - `'J'` → Joint Push Pull
  - `'V'` → Vector Push Pull
  - `'N'` → Normal 'Classic' Push Pull

- `selection` is a **list of valid drawing entities**, as an *Array*, preferably containing at least one face.

  Default: if **`selection == nil`**, then the current Sketchup selection is taken

- `distance` is the **distance**, positive for extrusion, negative for carving, as a *Sketchup length* (so usually passed as "20.cm or so). Must be non-zero.

  Be careful that for Joint and Normal Push Pull, the reference is the normal to the face, but for Vector Push Pull, positive would mean in the direction of the vector..

- `vector` is the **specifying a direction** as a *Vector3d object*
  - Joint Push Pull → indicate the privileged plane, by the vector normal to it
  - Vector Push Pull → vector of extrusion. Note that the orientation does matter
  - Normal Push Pull → ignored

  Note: must be a real vector created. Array like [x, y, z] won't work

- `finishing` is the **finishing option**, as a *single character string*,
  - `'D'` → Delete original faces
  - `'K'` → Keep original faces
  - `'R'` → Reverse faces for thickening

- `borders` is the **border option**, as a *single character string*,
  - `'N'` → No borders (just offset faces)
  - `'O'` → Borders on outer faces
  - `'A'` → Borders on all edges when generated faces are not coplanar

  Note: 'A' is applicable only to Normal Push Pull. It is taken as 'O' for other types of Push Pull

- `group` allows to **generate new faces as a Group or not**, as a *Boolean*

- `influence` specifies **whether the non-selected neighboring faces will influence or not the extrusion**, as a *Boolean*. Only applicable for Joint Push Pull.

- `angle` is the angle of Influence, as a *Floating number*, in degree (so 30.0 means 30 degree). Only applicable for Joint Push Pull.

RETURN VALUE

The method returns the status of the operation, as an ***integer***.

`Status = 0` if the operation was successful

Otherwise,

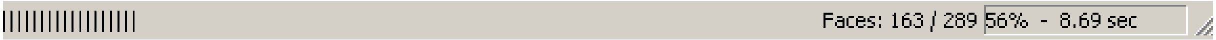
- `JPP_ERROR_InvalidParameters = 1`
- `JPP_ERROR_NoValidFaces =       2`
- `JPP_ERROR_InvalidDistance =    3`
- `JPP_ERROR_InvalidVector =      4`
- `JPP_ERROR_FailedExecution =    5`
- `JPP_ERROR_InvalidSelection =   5`
- `JPP_ERROR_OtherError =        99`

7. Last Word – Areas of Improvements

1) Performance

The macro is slow. It has to loop in the model to explore the topology, average vectors, compute positioning of textures, create faces, transfer properties of drawing elements, project holes, etc… In Ruby, which is an interpreted the language, this can take a fair amount of time.

I have implemented a **Progression Bar** to keep you aware of the progress of the operation, with some additional information on where it stands:



Faces: 163 / 289 56% - 8.69 sec

You may experience interruption of the progression. Don't panic, this is the Ruby garbage collector that is halting the process to clean up its memory. It should resume after a few seconds.

Likewise, if you click in Sketchup during a long operation, you may get a white screen. This is because the application is single-threaded, including the execution of Ruby scripts. Here too, the normal display should be restored at the end of the operation.

For the time being, I have found no way to interrupt the operation.

So bear with me this problem of performance!

2) Edge properties

I did not know too much what to do with the generated edges! The macro of course transfers properties such as layers, colors and shadow behaviors. It also transfers the 'Hidden / Visible" property, assuming that if the original edge is 'hidden', it is intentional. Then, the rules for deciding whether new edges should be or not smoothed and softened are based on some esoteric rules, the idea being to try to keep smoothing for internal edges, and remove it for external ones. But, even with this approach (rather complex to code), I am not sure you will obtain what you wish in all situations.

The only strict rule is that all diagonals resulting from triangulation of faces are smoothed and softened.

3) Compatibility

The macro does work in Sketchup v5 (as of v5.0.260 however). It can be a little bit slower because I had to emulate an API verb (Face.ClassifyPoint) that seems to be only available in version 6 (but version 5 being faster, you may not notice a difference!).

For Mac platforms, early adopters have advised that it works on this platform too.

4) Geometrical Issues

Occasionally, you may get unexpected results. This may come from a topological problem. But this can also come from bugs in the macro, as there are a very large number of geometrical situations. Thanks for feedback on the **Sketchucation forum**.