



# FreeCAD

## Thread for Screw Tutorial

### Introduction

This tutorial is a collection of techniques to model screw threads in FreeCAD. It was updated for v0.19, although the overall process has been essentially the same since v0.14, when the tutorial was originally written. The updated content focuses on the  [PartDesign Workbench](#) to create the thread, but does not use the  [PartDesign AdditiveHelix](#) tool as this was introduced later.

In traditional CAD systems modelling screw threads is discouraged because it puts a big load on the modelling kernel, as well as on the rendering of the shapes. In traditional systems a thread does not need to be represented directly in 3D space, as it can be indicated with its required characteristics in the 2D technical drawing that is sent for manufacturing. However, with the popularization of additive manufacturing (3D printing), there is now a real need to model 3D threads, in order to print them exactly as designed. This is what this tutorial is for.

Many of the techniques presented here have been collected from different forum threads:

- [Gathering thread modeling techniques \(https://forum.freecad.org/viewtopic.php?f=3&t=12593\)](https://forum.freecad.org/viewtopic.php?f=3&t=12593)
- [Creating a thread: Unexpected results \(https://forum.freecad.org/viewtopic.php?f=3&t=6506\)](https://forum.freecad.org/viewtopic.php?f=3&t=6506)

See also helpful videos:

- [Introducing a strategy for designing a bolt without the commonly found problems. \(https://forum.freecad.org/viewtopic.php?f=8&t=44259\)](https://forum.freecad.org/viewtopic.php?f=8&t=44259)

Remember that thread shapes take a lot of memory, and having just one thread in a document can increase the file size significantly, so the user is advised to create threads only when absolutely necessary.

### Method 1. Using utilities and parts from workbenches

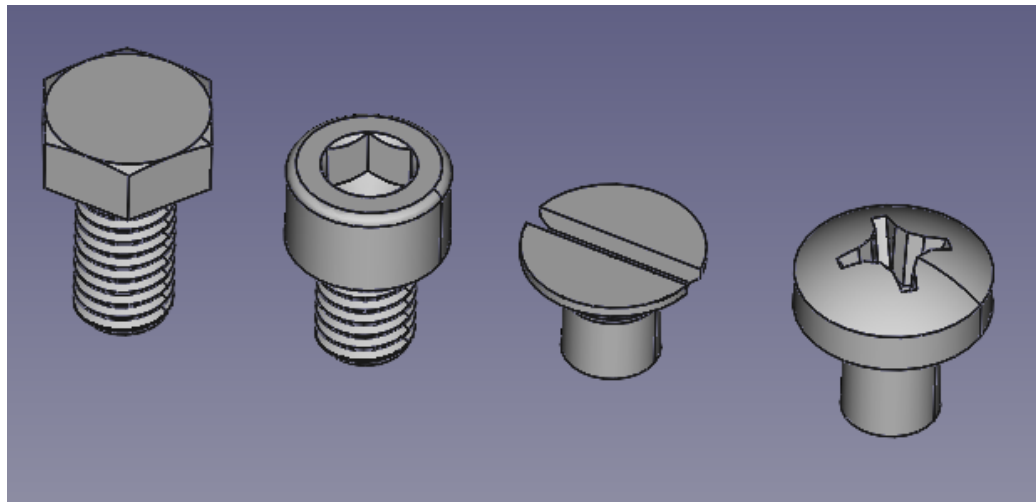
Using utilities and parts that other people have created is easy and saves time. See the [external workbenches](#) page for information on external tools.

In particular, three resources are recommended that can be installed from the [Addon](#)

 <b>Tutorial</b>
<b>Topic</b>
Product design
<b>Level</b>
Advanced
<b>Time to complete</b>
60 minutes
<b>Authors</b>
DeepSOIC, Murdic, vocx
<b>FreeCAD version</b>
0.19
<b>Example files</b>
Updated: Thread for screw tutorial ( <a href="https://forum.freecad.org/viewtopic.php?f=36&amp;t=44668">https://forum.freecad.org/viewtopic.php?f=36&amp;t=44668</a> )
<b>See also</b>
None

### Manager:

- Fasteners Workbench, to add/attach various fasteners to parts. The screws and nuts don't show a thread by default, but this can be controlled with an option.
- BOLTSFC Workbench, to place fasteners from the BOLTS library.
- ThreadProfile Workbench, to create common threads.



*Various standard screws inserted with the Fasteners Workbench. An option controls whether an object shows the real thread or just a plain cylinder.*

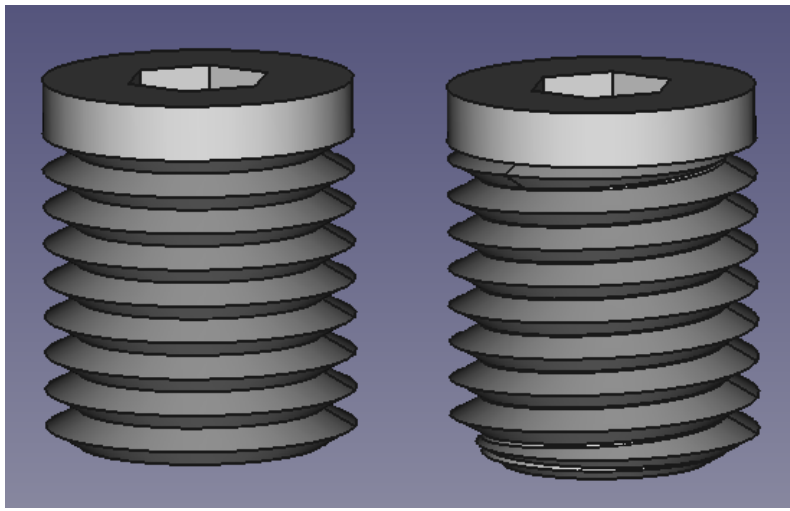
## **Method 2. Using macros (deprecated)**

- In the past, the Macro BOLTS was used to insert the parts from the BOLTS library. This is now deprecated. Use the BOLTSFC Workbench instead.
- In the past the stand-alone Screw Maker macro, by ulrich1a, was used to create individual bolts, screws, and washers. This is now deprecated. The Fasteners workbench, by shaise, includes the complete screw maker macro, together with a GUI to select the right component.

## **Method 3. Fake threads: non-helical**

In many cases we don't need real threads, we just need a visual indication that the threads will be there.

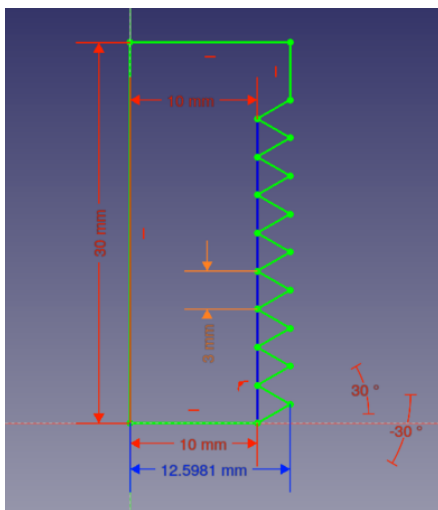
We can create a fake thread by using a non-helical path, for example by revolving a sawtooth profile, or by stacking discs with tapered edges. This fake thread is hard to tell apart from the real helical one by simple inspection. This method is good for visualizing a thread-like object, but it is not useful if we need to 3D-print an actual thread.



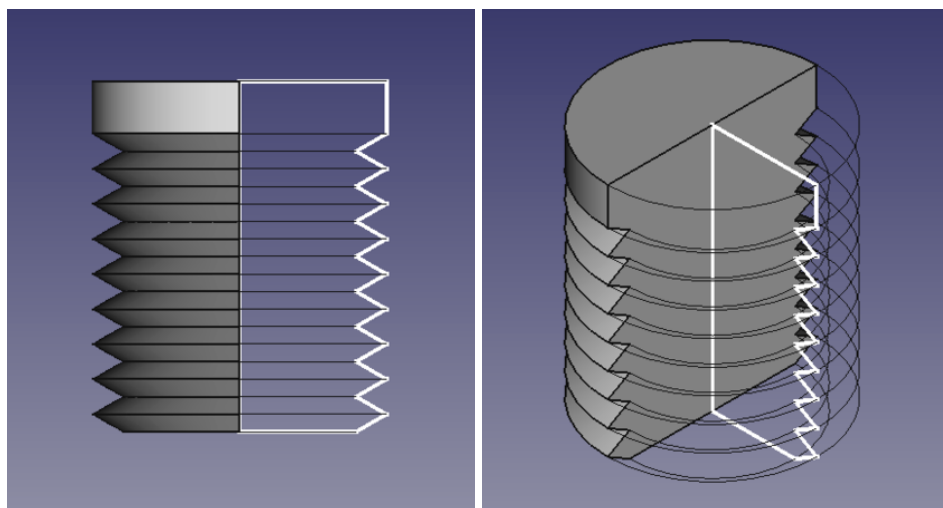
*Left: simple bolt with a fake, non-helical thread. Right: simple bolt with a real helical thread. When 3D printing is not needed, a simulated thread is often sufficient for visualization.*

## Revolving sawtooth profile

1. Click on **PartDesign Body**.
2. Click on **PartDesign New sketch**. Select XZ\_Plane.
3. Draw a closed sketch with the required inner diameter 10 mm, outer diameter around 12.6 mm, pitch 3 mm, number of teeth 8, and total height 30 mm.
4. Select the sketch, then click on **PartDesign Revolution**. Select Vertical sketch axis, and press **OK**.



*Profile used to create the revolution that will simulate a thread.*



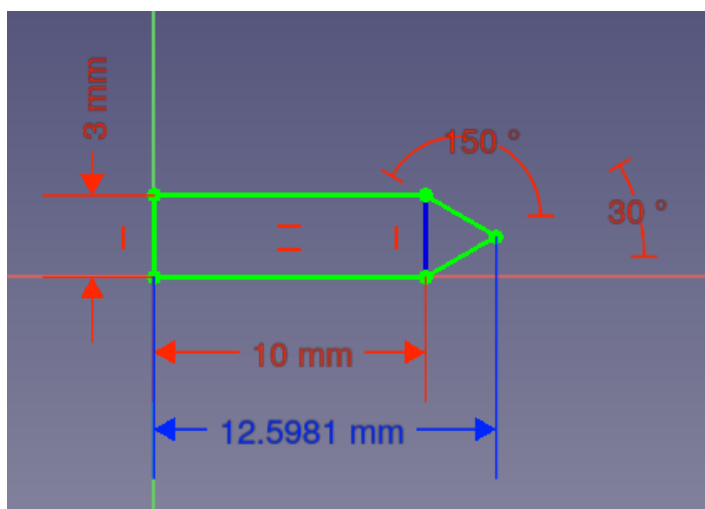
*Sectional view of the resulting non-helical thread produced by revolving the sawtooth profile around the vertical axis.*

## Stacking discs

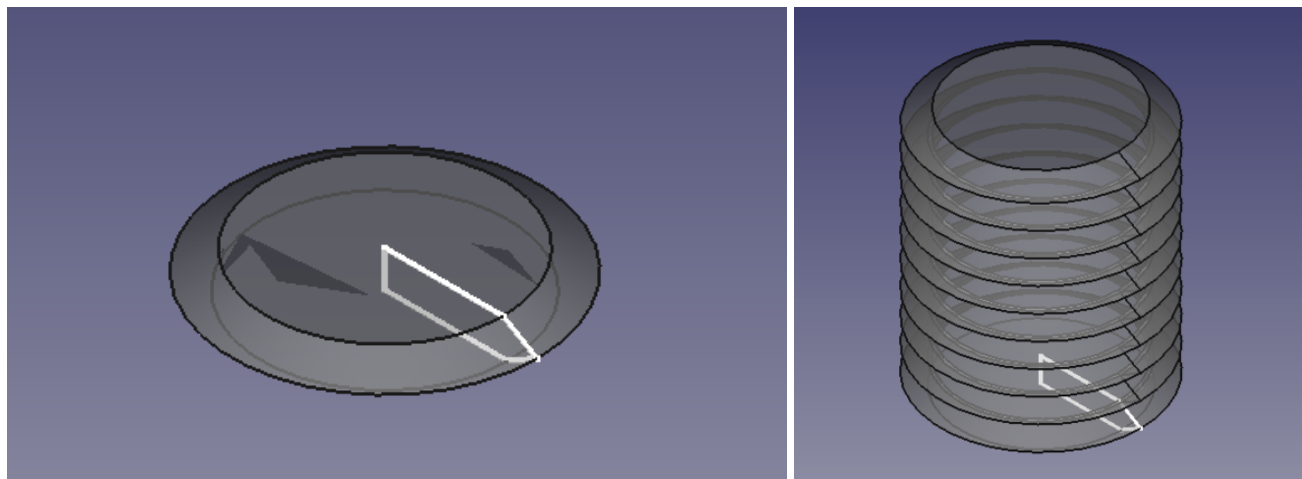
1. Repeat the first two steps from the previous section.
2. Draw a closed sketch with the required inner diameter 10 mm, outer diameter around 12.6 mm, and pitch 3 mm, but draw only a single tooth of the sawtooth.
3. Select the sketch, then click on **PartDesign Revolution**. Select Vertical sketch axis, and press **OK**.
4. Select the Revolution, then click on **PartDesign Linear pattern**. Select Vertical sketch axis. For a fake thread with a pitch of 3 mm, set the **Length** to 3, and **Occurrences** to 2, then press **OK**. This will create two discs, one on top of the other.
5. You can add more discs by increasing the value of **Occurrences** in the linear pattern, and by raising the **Length**, which is the total length of the fake thread.

The **Length** and **Occurrences** are related. If the length is too large, but the number of occurrences is not high enough, you will have disconnected discs, and the Body computation will fail, as the resulting object must always be a single contiguous solid. For example, to get a total height of 30 mm, set **Length** to 27 mm and **Occurrences** to 10.

If you wish, you may add a **PartDesign Additive cylinder** with a diameter equal to the inner diameter of the discs, and as high as the total thread height. This will join all discs into a single solid, thus guaranteeing that there will not be disconnected discs.



*Profile used to create a revolved disc that will be used to simulate a thread.*



*Left: single disc created by revolution. Right: multiple discs placed in a linear pattern in the Z direction simulating a helical thread.*

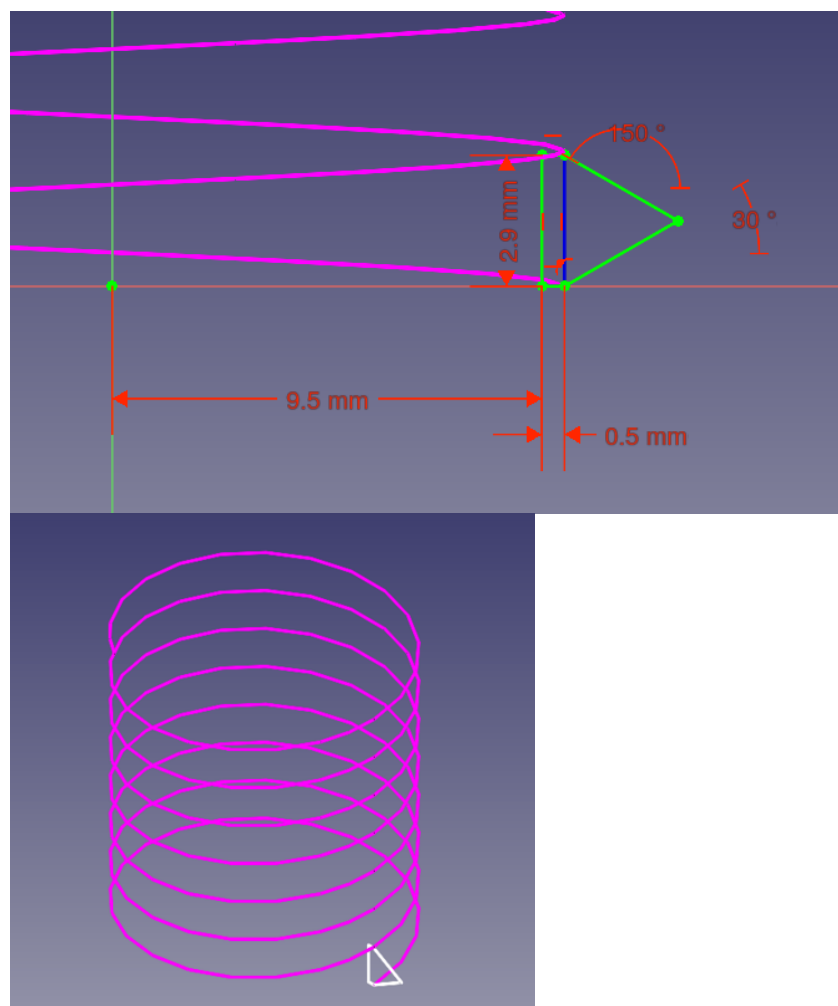
## Method 4. Sweeping a vertical profile

---

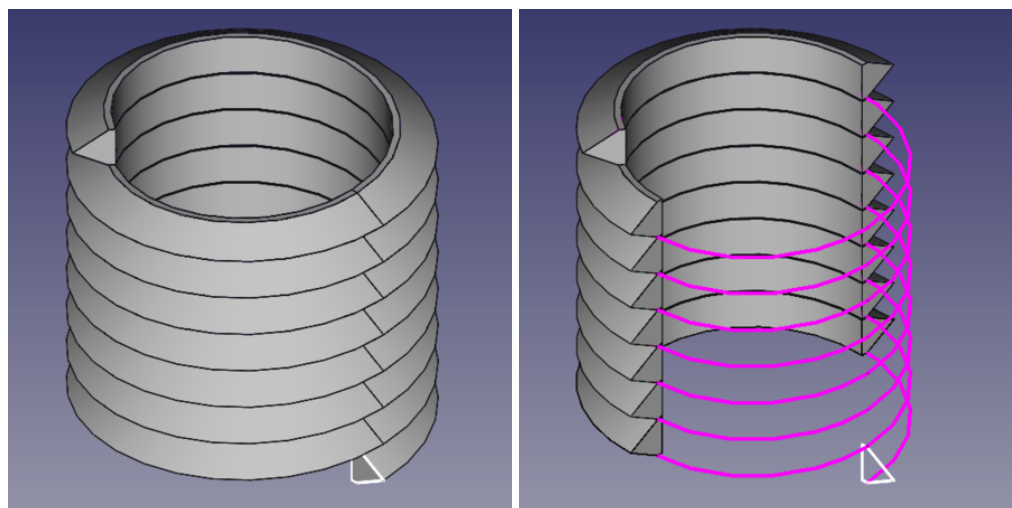
### PartDesign Workbench

A true thread consists of a closed profile sweeping a solid along a helical path.

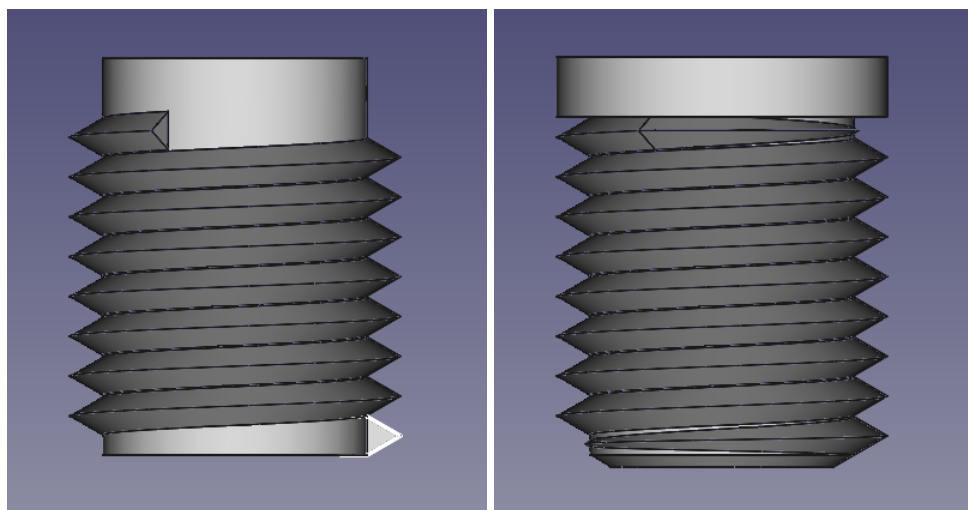
1. In the Part Workbench, click on Part Primitives to create a Part Helix. Give it the appropriate values for **Pitch** 3 mm, **Height** 23 mm, and **Radius** 10 mm.
2. Move to the PartDesign Workbench, and click on PartDesign Body.
3. Click on PartDesign New sketch. Select XZ\_Plane.
4. Draw a closed sketch with the required profile for the thread teeth, normally a triangular shape. In this case we will use a height of 2.9 mm, which is slightly smaller than the 3.0 mm pitch used for the helix path. The profile must not create any self intersections when moved along the helix, neither between the turns nor in the middle, thus the sketch as shown for stacking disks cannot be used.
5. Select the sketch, then click on PartDesign Additive pipe. In **Path to sweep along**, click on **Object**, and choose the helix object previously created. Then change **Orientation mode** to Frenet so that the profile sweeps the path without twisting; then press OK.
6. When the dialog asks for a reference, choose Create cross-reference.
7. The helical coil is created, but there is no central body or shaft.
8. Click on PartDesign Additive cylinder with the appropriate **Radius** 10 mm and **Height** 29.9 mm to touch the rest of the helical thread and automatically fuse to it.
9. Additional boolean operations are needed to shape up the abrupt ends of the coil. For example, you can use additive features to provide a head to the screw, and a tip.



*Left: profile for a helical thread. Right: helical path that will be used to create a sweep.*



*Left: helical coil resulting from the sweep operation of the closed profile along the helical path. Right: sectional view of the coil produced from the sweep.*



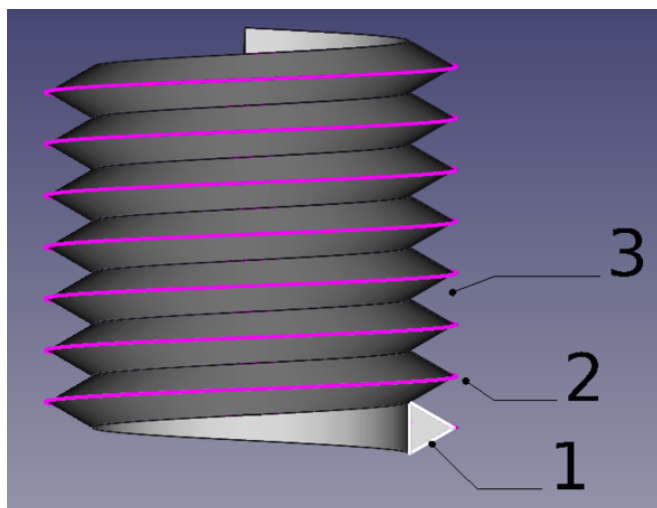
*Left: helical coil fused to a central cylinder to form the body of the screw. Right: more features, a head and a tip, added to improve the shape of the screw.*

## Part Workbench

This process can also be done with the tools of the Part Workbench.

1. In the Part Workbench, click on Part Primitives to create a Part Helix. Give it the appropriate values for **Pitch** 3 mm, **Height** 23 mm, and **Radius** 10 mm.
2. In this case, you don't need a PartDesign Body. Switch to the Sketcher Workbench, then click Sketcher New sketch, and choose the global XZ plane.
3. Then return to the Part Workbench, and use Part sweep.
4. Select the appropriate sketch from **Available profile** and click the arrow to pass it to **Selected profiles**.
5. Click Sweep path, and choose all edges of the existing helix in the 3D view. Click Done.
6. Make sure to tick ☒ Create solid and ☒ Frenet. Obtaining a solid is the key to be able to perform Part Boolean operations with the resulting coil, otherwise only a surface will be produced.
7. Click OK to exit the dialog and create the coil.


Now you can proceed to add other primitives like Part Cylinders, or other shapes, in order to perform Part Fuses and Part Cuts to finish building the screw.

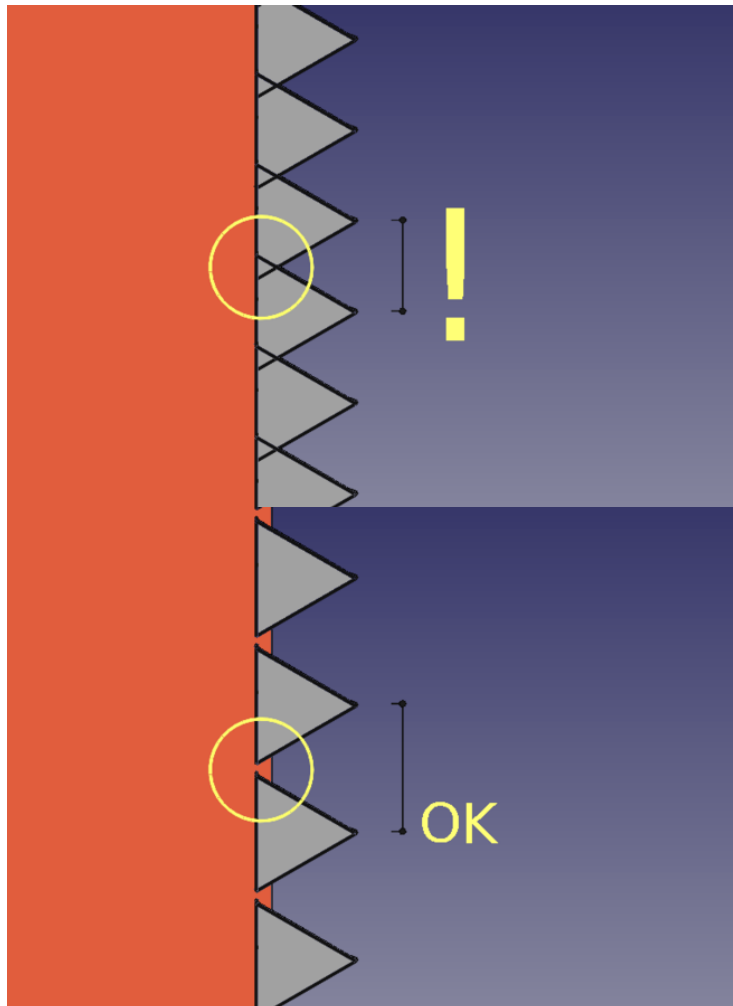


*Creating a thread coil by sweeping a vertical profile, (1) the sketch profile, (2) the helical sweeping path, and (3) the result of the sweep.*




## Tips for success

- **Rule 1.** When the profile sweeps the helix, the resulting solid coil must not touch or self-intersect as it will be an invalid solid. This holds for the profile moving along the helix, as well as intersections in the center of the helix. Attempts to do boolean operations with it (fuse or cut) are very likely to fail. Check the quality of the coil with  `Part CheckGeometry`; if self-intersections are reported, you must increase the pitch of the helix.

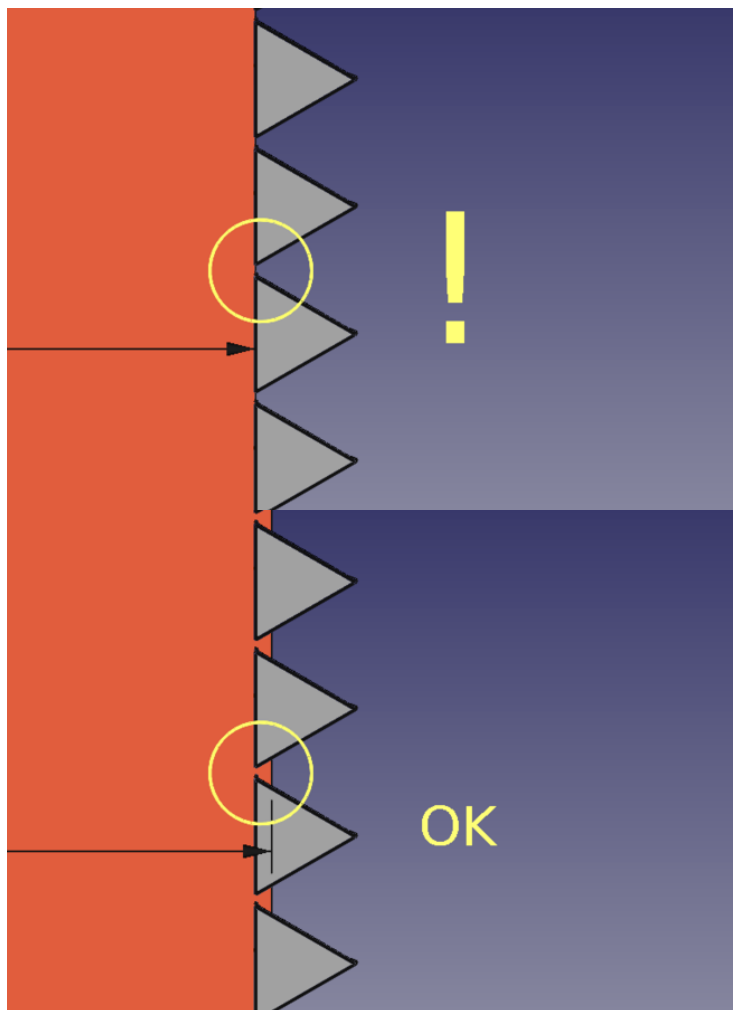


*Left: invalid sweep generated by using a very small pitch of the helix compared to the height of the triangular profile. Right: pitch that is sufficiently large and doesn't cause self-intersections.*

- **Rule 2.** When a cylinder is added to a coil to form the main shaft of a screw, the cylinder must not be tangent to the coil profile. That is, the cylinder must not have the same radius as the inner radius of the thread, as this is very likely to fail a fuse operation. In general, avoid geometry coincident to elements of the sweep, such as tangent faces, or edges tangent to faces they are not connected to. In order to produce a good boolean union, the swept coil and the cylinder must intersect. Check the quality of the fusion with  `Part CheckGeometry`; if coplanar faces are reported increase the cylinder's radius by a small amount.
- If the coil and the cylinder are tangent, even if the first fusion succeeds, it may fail in subsequent boolean operations with a third solid.
- This is a limitation of the OpenCASCADE Technology (OCCT) kernel; in



general, it doesn't handle well operations between coplanar surfaces.



*Left: the solid cylinder is tangent to the inner radius of the thread; this may result in an invalid boolean union. Right: the cylinder has a slightly larger radius, so the two solids intersect; this will result in a valid fusion.*

- **Rule 3.** The inner cylinder has a seamline. You should avoid placing the start of the helix along that seam. Either turn the helix or the cylinder by some degrees.
- **Tip 1.** The radius of the helical path does not matter, unless the helix is tapered. All that matters is the pitch and the height of the helix. This means that you can use a single `Part Helix` to generate several threads with equal pitch. What determines the position of the resulting coil is the position of the profile sketch.
- **Tip 2.** Keep the thread short, that is, with a low number of turns. Long threads tend to fail with boolean operations. If you need to add many turns, consider creating first a short thread, and then using `Draft OrthoArray` to duplicate the same pattern several times.
- **Tip 3.** For 3D visualization and 3D printing it may be okay to leave the cylinder and the thread unfused, that is, with intersections between the two solids. Reducing the amount the boolean operations results in less memory

consumption and smaller files.

### Pros and cons

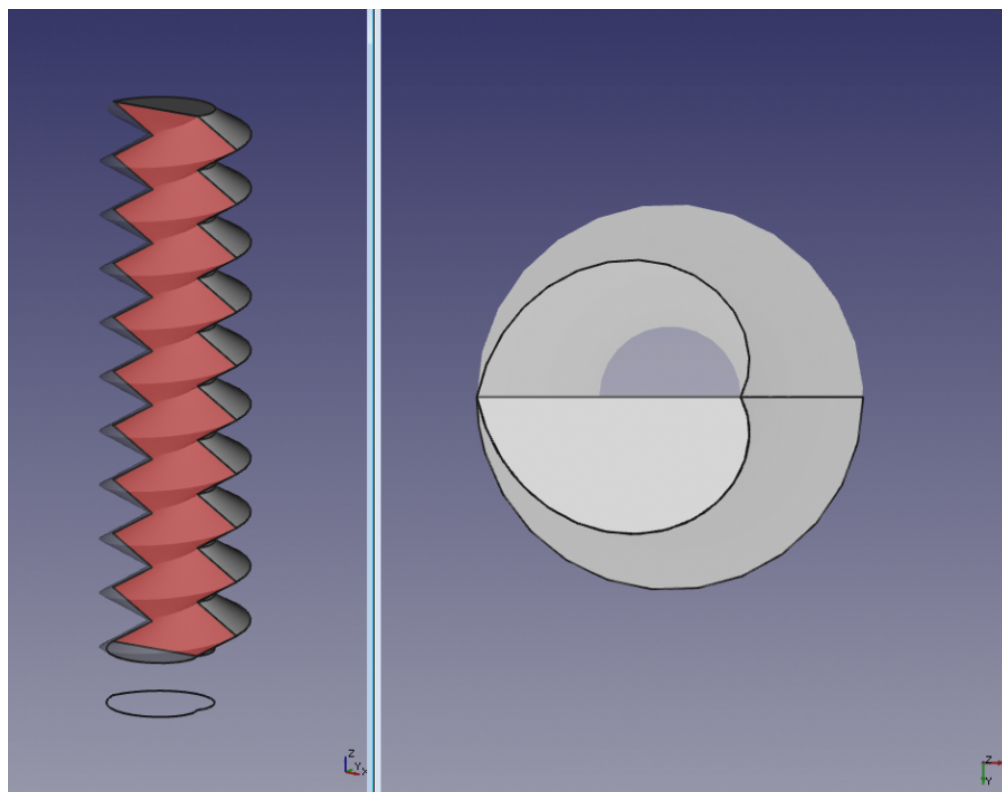
- ✓ Easy to understand.
- ✓ Very natural way of defining a thread profile.
- ✓ No problems with meshing of the resulting object, unlike method 5.
- ✗ Due to invalidity of self-intersecting sweeps, it is next to impossible to generate a thread with no gap between each tooth, that is, with no straight cylindrical face at the inner sides of the thread.
- ✗ Boolean operations are required to obtain a single contiguous solid. Boolean operations take a relatively long time to calculate, and fail often.
- ✗ Threads with a high number of turns are problematic.

## Method 5. Sweeping a horizontal profile

---

### General

The idea is to sweep a horizontal cross-section of the thread along a helix. The main problem here is figuring out the profile to use to obtain such thread.



If one uses a circle as a horizontal profile (the circle has to be placed off the origin, that offset defines the depth of the thread), the thread profile will be sinusoidal.

To obtain a standard sawtooth profile, a pair of mirrored archimedean spirals need to be fused into a wire. The resulting figure is a heart shape, which becomes barely distinguishable from a circle when the depth of the thread is small compared to its diameter, this is why a "thick" thread is shown on the picture above.

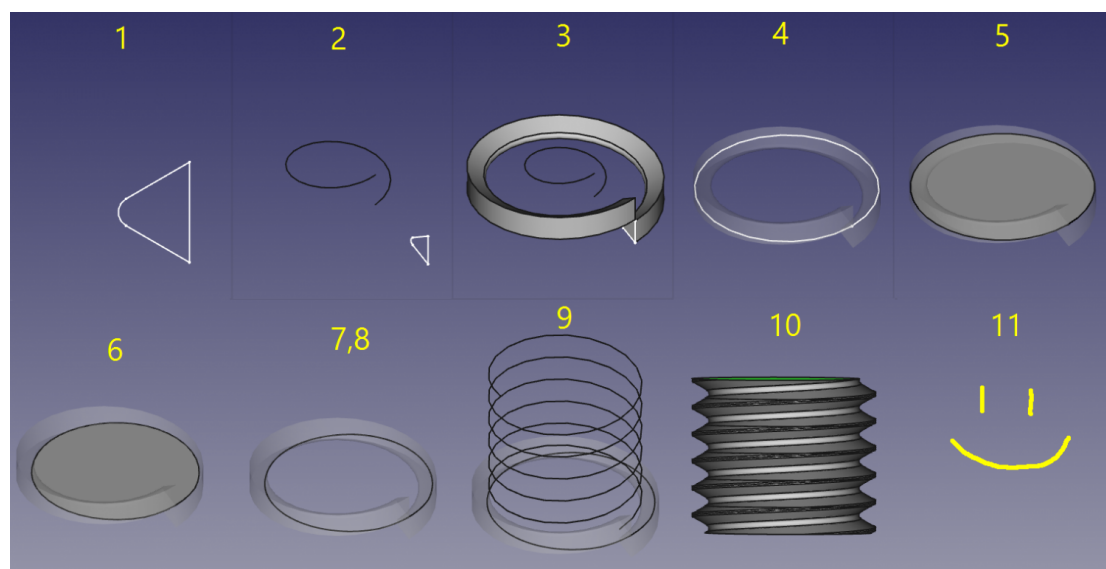
## Generating the profile

Figuring out the horizontal profile to obtain a certain vertical profile is not easy. For simple cases like triangular or trapezoidal it can be constructed manually. Alternatively, it can be constructed by creating a short thread with method 4, and getting a slice of it by doing a Part Common between a horizontal plane face and the thread.

### Profile for triangular thread

1. First create an Archimedian spiral in the XY plane.
  1. Set the number of turns to 0.5.
  2. Set the radius to the inner radius of the thread, the outer radius will be this plus the depth of the cut.
  3. Set the growth to double the depth of cut of the thread.
2. Part Mirror the spiral against the XY plane
3. Part Fuse the spiral and the mirror to obtain a closed wire, shaped like a heart.

### Profile for arbitrary cross-section



1. Make a vertical cut profile. Make sure that the height of the sketch matches the pitch of the thread you need.
2. Make a helix1 with height identical to the pitch and the pitch identical to the thread pitch, and a helix radius of  $0.42 \times \text{nominal diameter of the thread}$ .
3. Sweep the cut profile along the helix1. Set ☒ Create solid and ☒ Frenet.
4. Make a circle with nominal radius of the thread in the XY plane.
5. Make a face from the circle. This can be done with  or , then set Data **MakeFace** to true.
6. Cut the face with the sweep profile.
7. Make a  from the cut.
8. Use  on the clone in order to get a wire. This wire is the horizontal profile needed for this method.
9. Make a helix with radius of nominal radius of the thread and a pitch of the thread and the height of the needed thread.
10. Sweep the wire along the helix. Set ☒ Create solid and ☒ Frenet.
11. You are done.

The step-by-step guide was taken from this forum post by Ulrich1a (<http://forum.freecad.org/viewtopic.php?f=3&t=6506#p52558>) ("Creating a thread: Unexpected results"), slightly

modified.

The steps are also shown in action on [this video](http://www.youtube.com/watch?v=fxKxSOGbDYs) by Gaurav Prabhudesai (<http://www.youtube.com/watch?v=fxKxSOGbDYs>) ("FreeCAD : How to make threads").

### Pros and cons

- ✓ a ready-to-use thread-on-a-rod solid shape is created by the sweep directly.
- ✓ fewer or even no boolean operations are required, so generation speed is very high compared to Method 4.
- ✓ thread ends are nicely cut straight away
- ✓ long threads are not a problem, unless a boolean operation is needed. Otherwise, it is not going to be much better than Method 4.
- ✓ threads without a gap are not a problem.
- ✗ defining thread profile is complicated.
- ✗ using the standard mesher with a thread created in this way generates ugly meshes, which can lead to problems. Other meshers are better, for example, Mefisto seems to give the best results.
- ✗ large memory footprint according to [Gathering thread modeling techniques](http://forum.freecad.org/viewtopic.php?f=3&t=12593&start=10#p101197) (<http://forum.freecad.org/viewtopic.php?f=3&t=12593&start=10#p101197>).

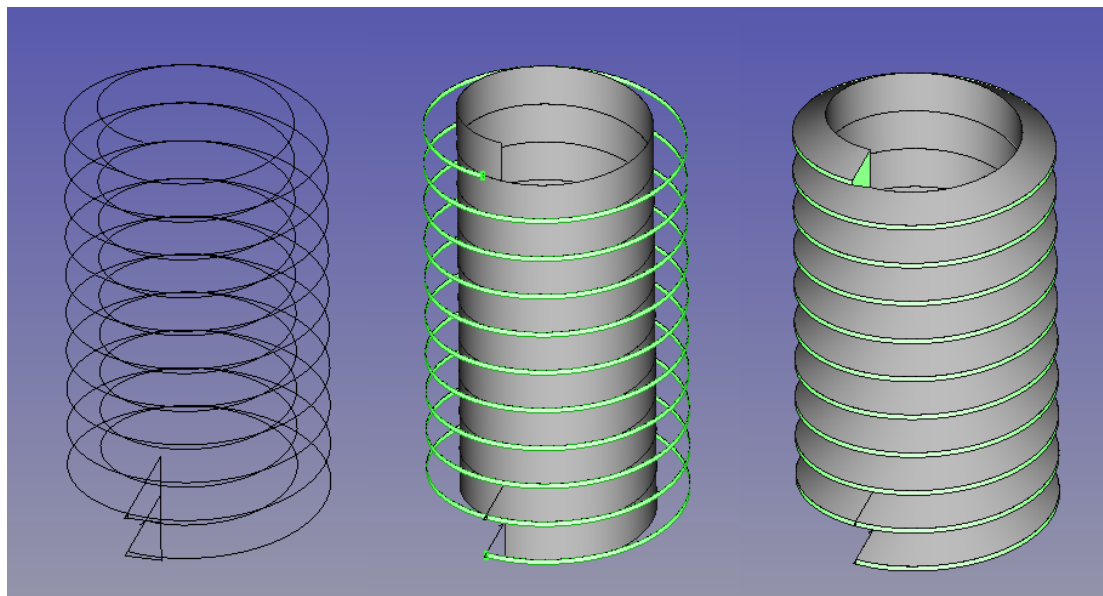
## Method 6. Lofting between helical extruded faces

### General

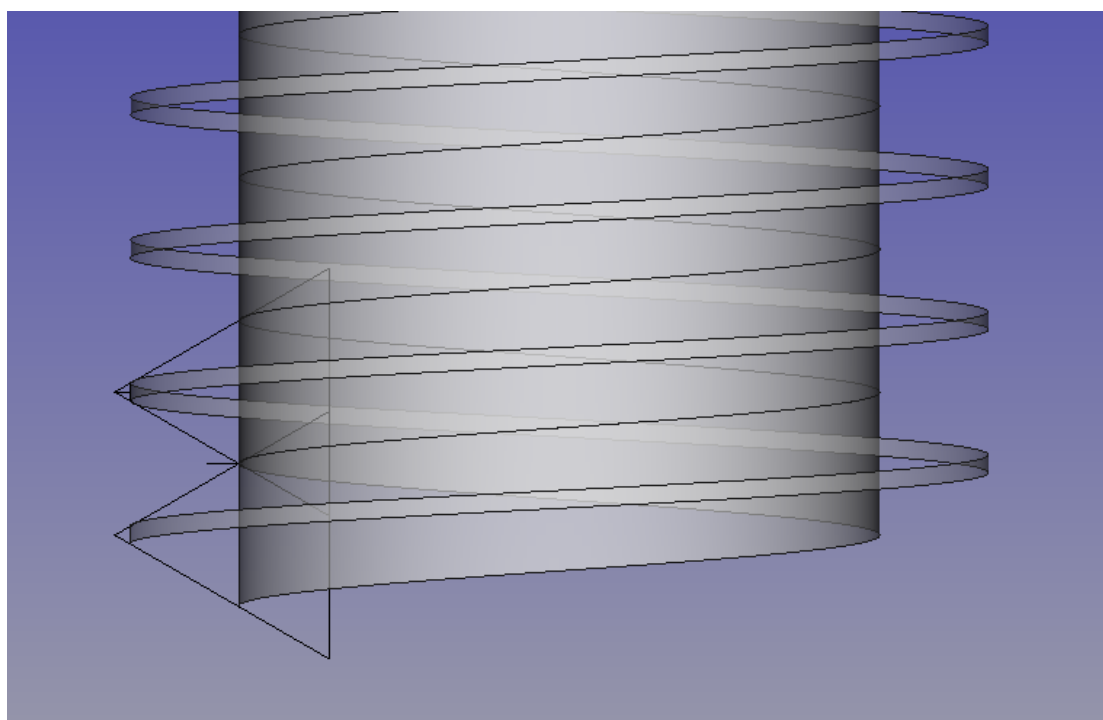
Helical splines will extrude coaxial faces that are able to be lofted, while FreeCAD's parametric helix won't. It takes two helical splines to define a thread. Those two can be scaled from a library spline, then located and extruded appropriately to get the form right.

FreeCAD's parametric helixes aren't truly helical, but helical b-splines aren't difficult to lay out. One manual method is to array dodecagons (12-sided polygons) with 5mm radius/10mm diameter at 1/12mm (0.08333.mm) z intervals and trace splines from vertex to vertex in ascending and rotating order, and to consider doing it once with, say, 10 turns, so that that spline can be re-used as a library file for import and reuse. It's convenient to use 10mm diameter/1mm pitch for ease of scaling. If you are doing it manually, drawing a Dwire and then converting it to a b-spline is easier than drawing a spline. Dwires don't have curvature computed while being drawn, so they follow the cursor and snap more obediently.




Once the splines are scaled to the right size and located so that the loft will have the right included angle between the thread flanks, they're extruded along their axis, a pitch length's worth for the inner spline, the outer pitch/8.



ISO and other threads have relieved, ie flat, inner and outer edges rather than sharp, which suits FreeCAD users with this method, because we can loft to the helical face at the nominal fastener size, while an inner face can't be lofted to an outer edge spline because a face is a closed profile, a spline is open. ISO standard says the nominal size of external threads have a face width  $\text{pitch}/8$ . The picture shows how the geometry is arranged, and the helical faces that result. Then, loft between the faces, and then a cylinder that gives the inner helical face, which ISO puts at  $\text{pitch}/4$  width, is added to the threads.



This method produces reliable solids that boolean properly. While it doesn't produce "parametric" screw threads in standard sizes in the sense of having simple access to form by fastener size, it's an easy way of producing an accurate library for reuse, and models of specialised forms like ACME, or Archimedian screws, are also uncomplicated as one-offs.

 <a href="#">PartDesign</a>	<a href="#">[Fold ud]</a>
 <a href="#">Sketcher</a>	<a href="#">[Fold ud]</a>
 <a href="#">User documentation</a>	<a href="#">[Fold ud]</a>

Hentet fra "[http://wiki.freecad.org/index.php?title=Thread\\_for\\_Screw\\_Tutorial/en&oldid=1355784](http://wiki.freecad.org/index.php?title=Thread_for_Screw_Tutorial/en&oldid=1355784)"