

Preparing Rhino files for 3D Printing

Index:

- General considerations
- Objects check and validation
- Naked edges
- Converting NURBS to meshes
- NURBS to mesh Detailed controls
- Accuracy
- Related Articles

General Considerations.

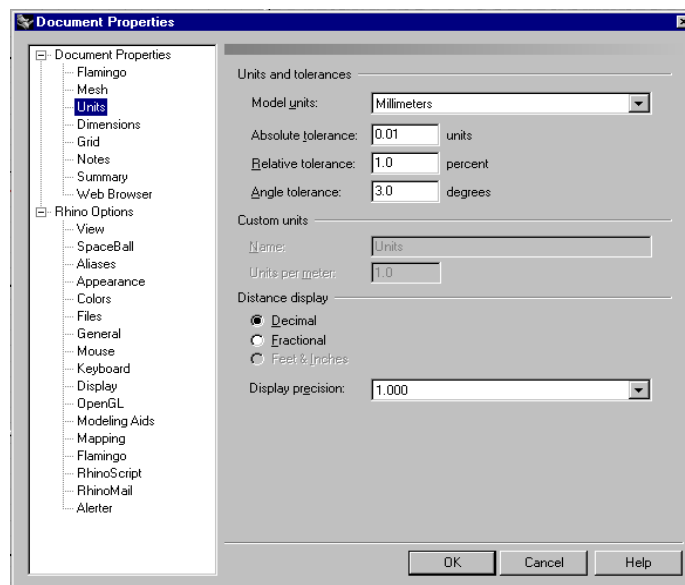
In this guide we will see all concepts related to file creation, analysis, repair and export, when files are aimed for 3d print systems.

Before start modeling anything using Rhino there are some items that should be considered.

If the model is going to be exported to other 3d programs that supports NURBS, and we are going to use IGES or STEP file export formats (both support nurbs), you must know, before starting to model, what file tolerance must be used to be able to export to the specific program you will be exporting to.

Then use a tolerance 10 times tighter than that (e.g. If the receiving Software works with 0.01 mm tolerance then set Rhino to 0.001 tolerance before starting to model).

Rhinoceros file tolerance can be set from document properties window, Units tab.



This ensure us that different surfaces that compose a Polysurface were created within guest program tolerance, so it can understand them as stitched (if we use STEP) or allow them to be joined later in the receiving software if we use IGES.

Another factor we should consider before starting to model in Rhinoceros is knowing object size and tolerances used for its later production. If this tolerance is lower than needed to export the model then you must adjust it if necessary.

Verify

The 3d model to export should not have naked edges.
The 3d model to export should not be a bad object.

Troubleshoot

Detect and close naked edges by fixing the surfaces that create naked edges or create new ones for filling the holes. An object with naked edges cannot be printed in 3d, because it is creating an open mesh when exporting it, and 3d print systems need watertight meshes for printing.

If a bad object is detected (Checknewobjects or Selbad commands), explode it, use again Selbad command to detect the specific bad surfaces and delete them.

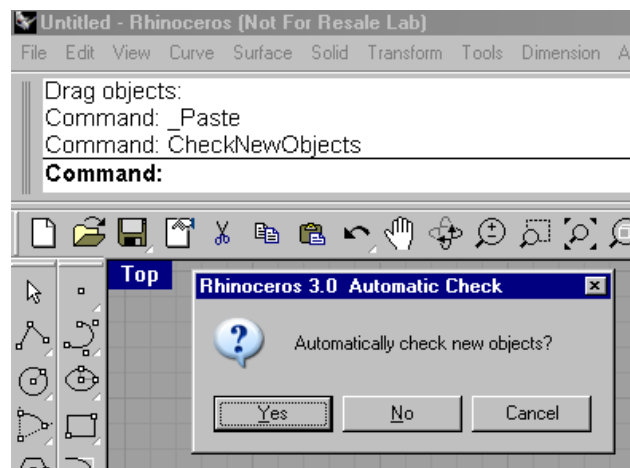
Use Rebuildedges command in surfaces to make sure there is no forced edge and recreate the deleted bad surface.

Verification and object validation

Before starting to model, and if you have done this previously run CheckNewObjects command by typing it in the command prompt

This command will check in real time, every geometric figure created to verify if it is valid (not bad) and pops up a message when the figure created is corrupted.

Advantages of using "CheckNewObjects" command: -It advises us when bad geometric figures are created. -it is always ON even between sessions allowing us to instantly detect a bad object when it is created, and give us the possibility to "undo" what we have done and check our procedures.

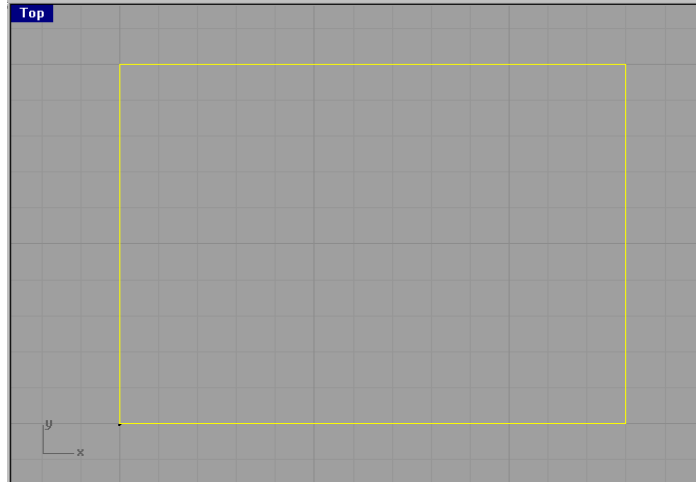


Bad geometry could be caused for many reasons, but an easy repeatable example is creating a rectangle with one corner open. The gap should be smaller than Rhino's tolerance. This can be achieved by moving its control points until you place line ends too close.

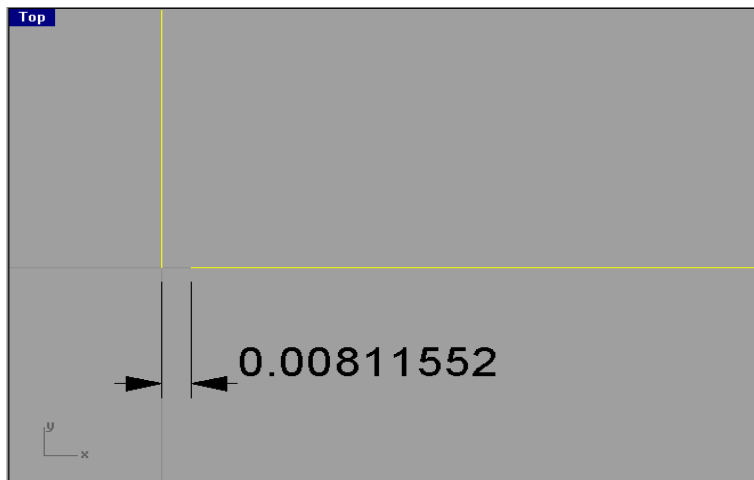
Using Osnap prevents these errors.

If Rhino's tolerance setting is set to 0.01, then make the gap as small as 0.005. This will allow us to create an open curve, but with its aperture smaller than Rhino's tolerance.

At first sight, it will look like a closed rectangle.



But if you zoom in you will be able to "see" the gap. The gap distance which is 0.008, is way smaller than Rhino's tolerance, 0.01.



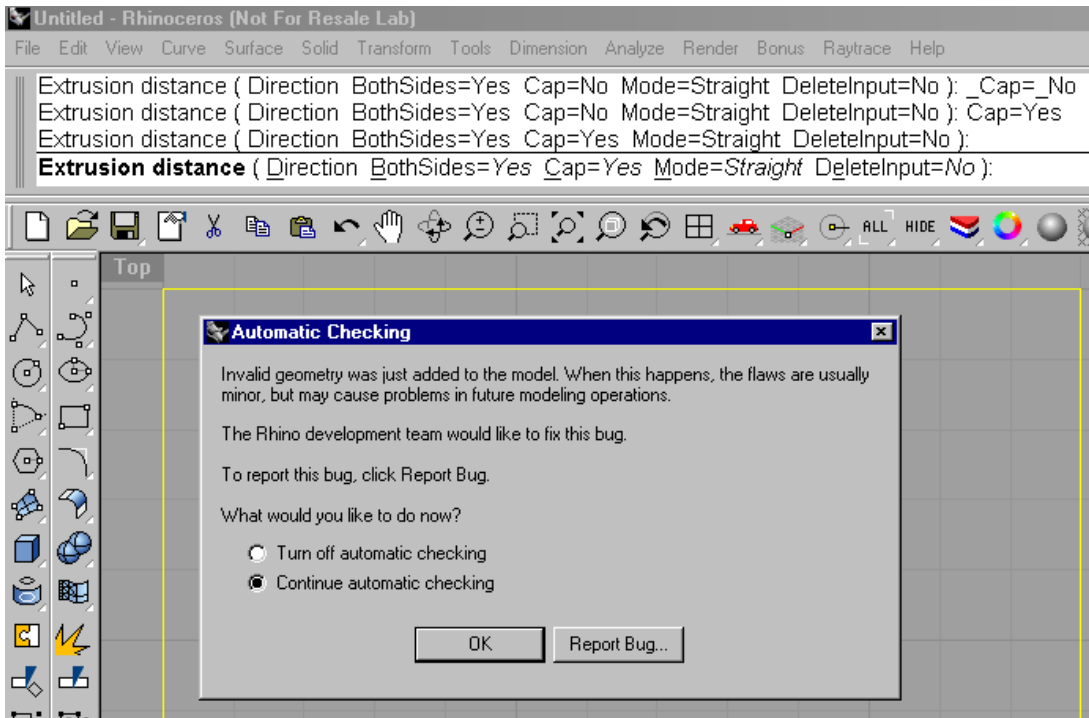
When an open curve is extruded, Rhino won't create its caps, not even if Caps option is set to: Yes.

But, if we extrude this open curve and set Caps option to: Yes, Rhino will try to create them. Because it cannot read the curve as open, the gap is too small, and it will consider the curve as closed.

Rhino will try to create caps from an open curve and generate a **BAD OBJECT**.

We should never allow bad objects to be created. Bad objects should be eliminated.

If CheckNewObjects is active it will tell us automatically when a bad object is created with a pop up window. When the window shows up we have the option to continue detecting bad objects or turn CheckNewObjects command off. The usual choice is to continue detecting.



Another advantage of using this command is that once it is activated it remains active, even if we close Rhino and open a new session.

This command also checks geometry when a Rhino file is opened.

This will also allow us to detect bad objects in the exact moment that they get created and use undo to do one step back and check our geometry.

To be able to print our model in 3D printers we must check for bad objects and find none, and then also check for naked edges. If the object has naked edges it cannot be 3d printed.

Naked Edges.

What does it mean when an object has naked edges? Well, basically it means that it is open. Well modeled objects should not be open.

It means that some surface edges aren't joined to other surface edges. Some surface edges are touching air instead of other surface edges.

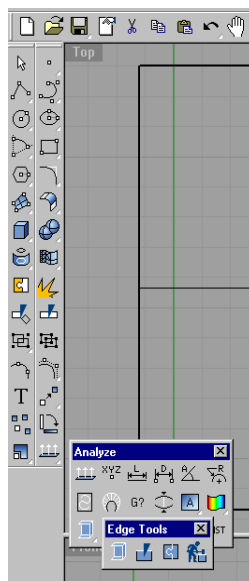
Naked edges are not a problem if the model will be used for render, however they aren't acceptable for CAM or 3d print.

A model with naked edges cannot be 3d printed.

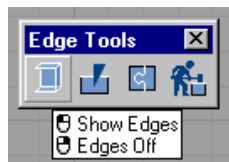
How can we detect a naked edge?

When we have a simple model it is easy to detect naked edges, but when the model becomes more complex we must use Rhino's analysis tools

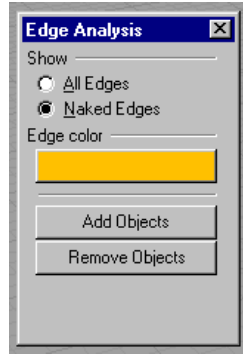
Inside Analysis toolbar, we will find several tools, some very important and not very well known by users.



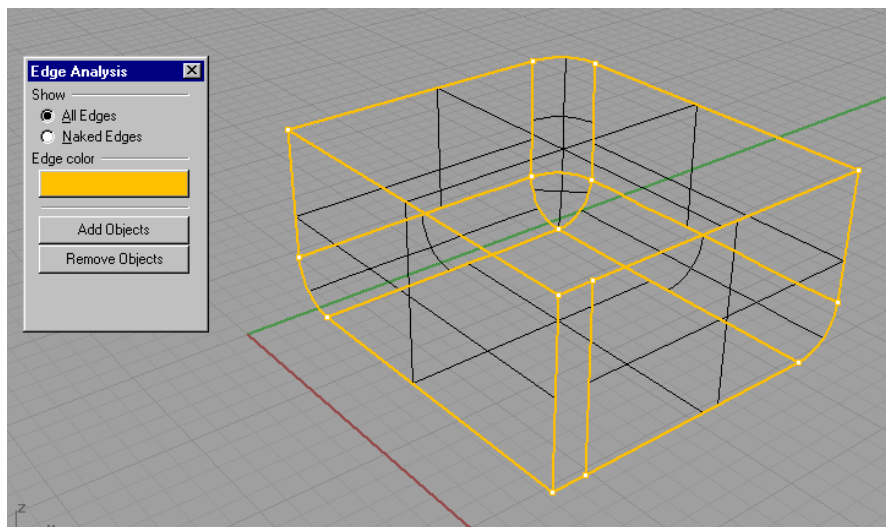
The first icon has 2 functions: "Show edges /edges off". When running the "Showedges" command (if nothing is previously selected), the command asks us to select the object to analyze and then press Enter (or Right mouse button).



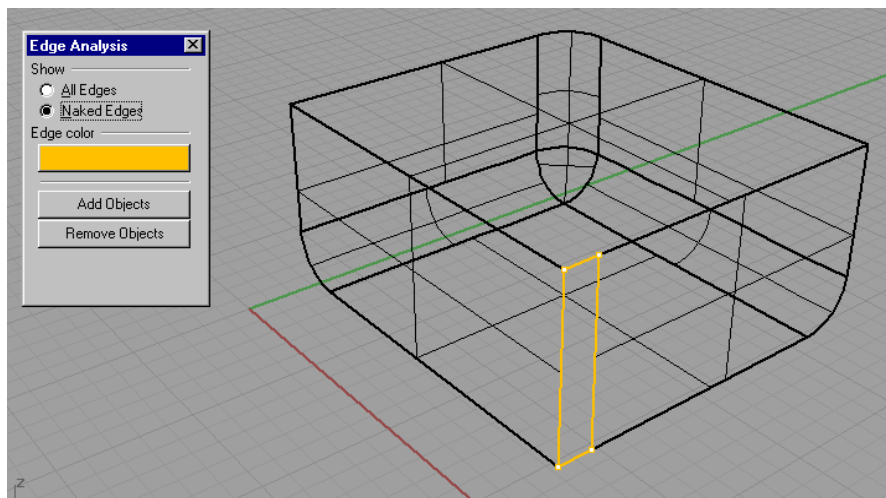
A new “edge analysis” window will show up with 2 options “All edges” and “Naked edges”.



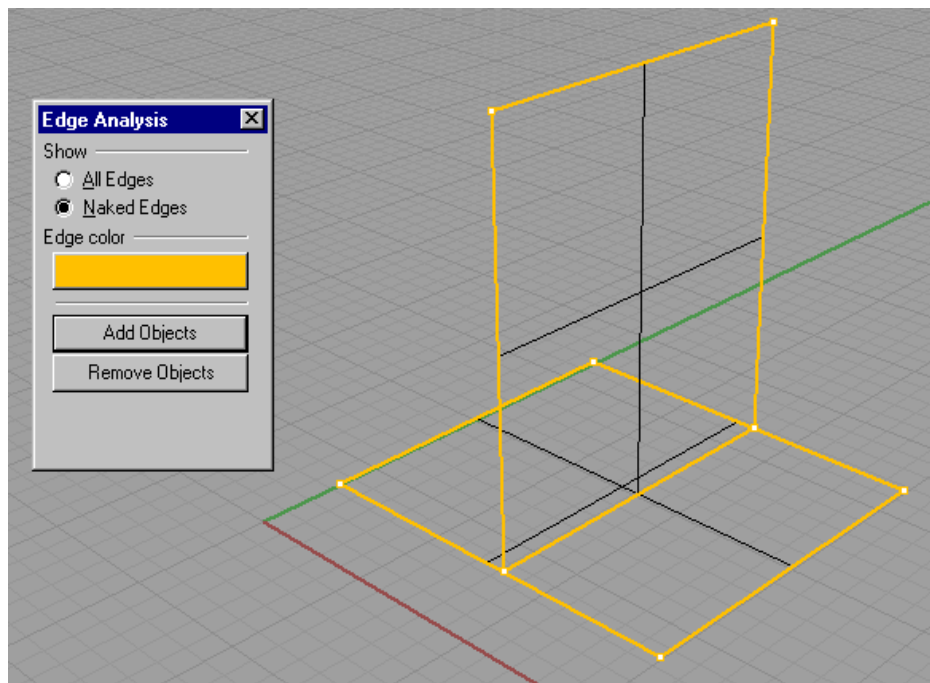
If we choose “All Edges” option, Rhino will show in the chosen color in “edge color” all surfaces edges (including seams).



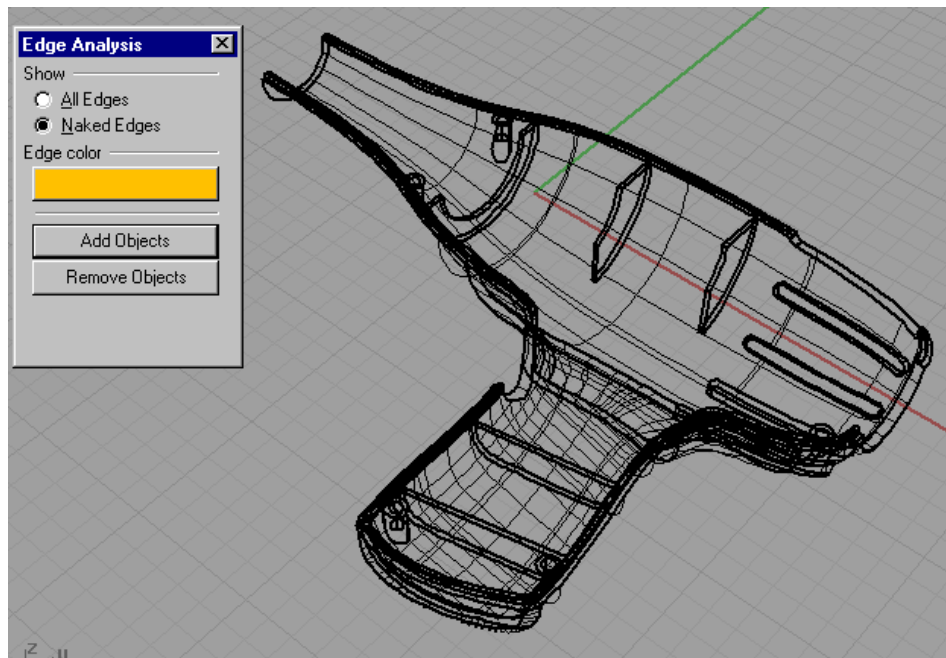
If we choose “Naked edges” only edges of surfaces not joined will be shown in color.



It is necessary to clarify those surface edges that are in touch with other surface edges (but not joined) or in touch with other surfaces in a T shape will be shown as naked edges. If surfaces cannot be joined, even if they are in “touch” the edge will be shown as naked.



Naked edges shows in the model where it is open and for that reason wrong.
A part of any material cannot be represented by an open Polysurface.
Hollow pieces must be modeled with exterior and interior surfaces and no naked edges.



When a naked edge is detected there are several strategies to “close” these open Polysurfaces.

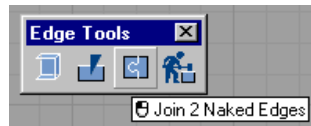
If naked edge analysis is showing a missing surface, then the solution is relatively easy, just create the missing surface and join it to the rest of the model. That should solve the problem.

Sometimes naked edges are harder to fix, usually caused by a not very clean modeling technique.

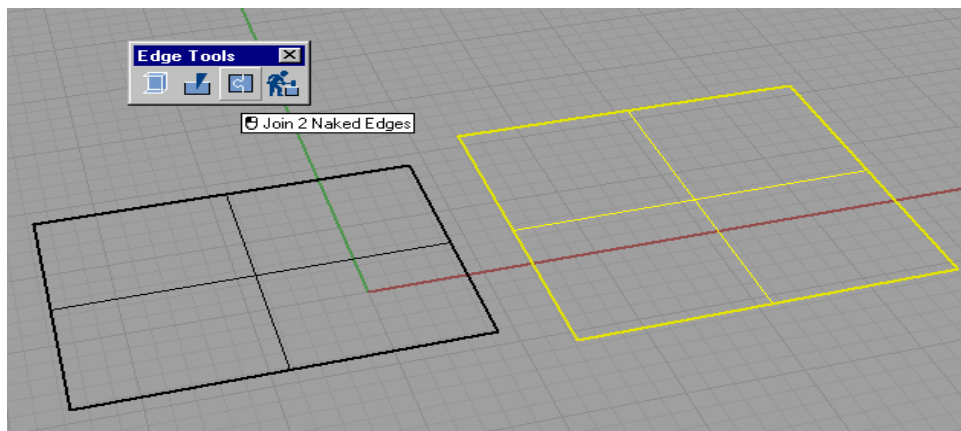
Sometimes is it caused by Osnap incorrect use when placing curves that will generate a surface later on. Or, using surface commands like Patch, without using necessary parameters to get the precision required in patch edges. Creating polylines superposed, and many other reasons.

If precision modeling is not an important factor and you need a quick fix to get the model printed in 3d, there is a command that can be used to create a “forced” union.

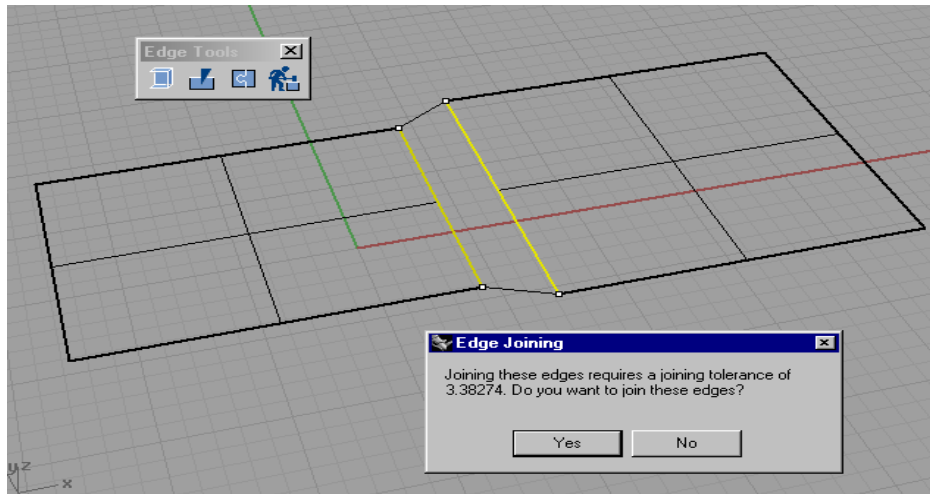
A forced union is necessary when join does not work because surfaces edges are more distant to each other than Rhino tolerance.(e.g. when 2 surface edges are 0.1mm apart and Rhino tolerance is 0.01, join command wont work). JoinEdges edge command will force this union.



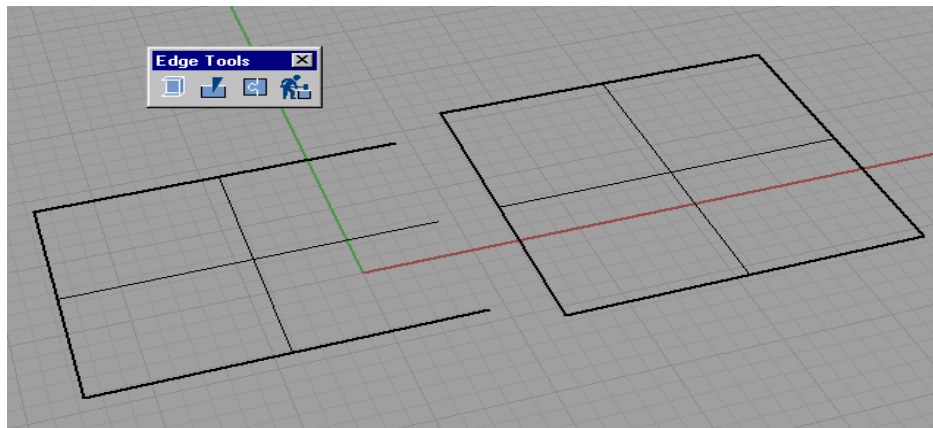
This command will force the union, because it is not paying attention to Rhino tolerances.



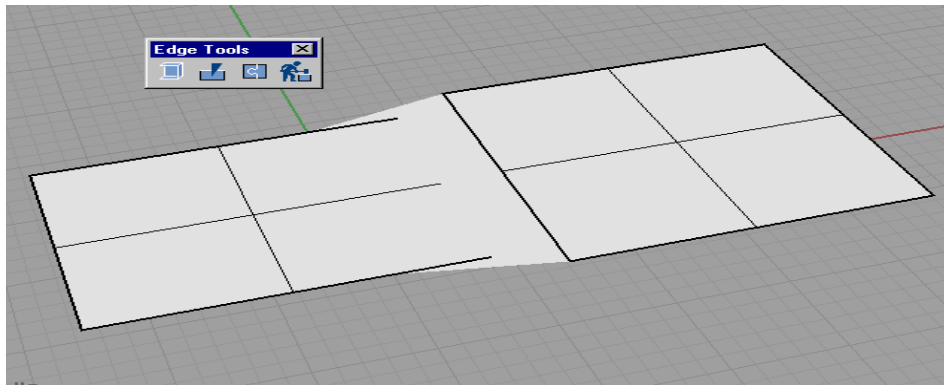
When using Join edge command a message will appear telling us that surface edges will be joined out of tolerance.



You can see in the resulting surface that one edge seems to disappear. It has been moved to the other edge position.



When using shaded viewport we can see how shade mesh is created to show surfaces as joined.

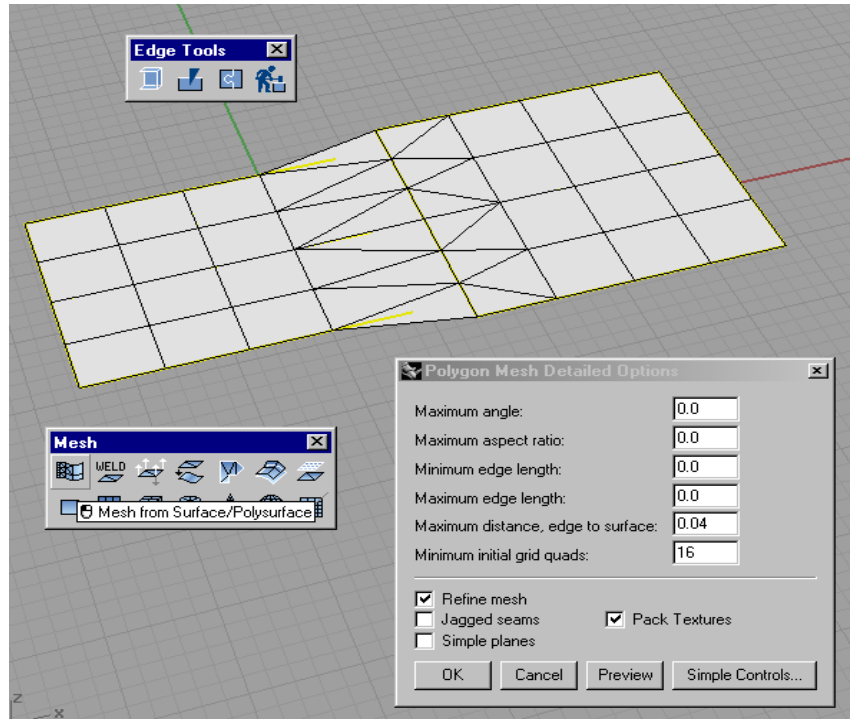


To verify this we can create a mesh from these NURBS surfaces.

We can see how the mesh is created with a perfect union.

This mesh is the same mesh type that is created when we export the NURBS model in mesh format, the model is converted to a mesh file and the mesh file is what is exported.

STL file format is the one recommended for 3d print, and it is a mesh only format.

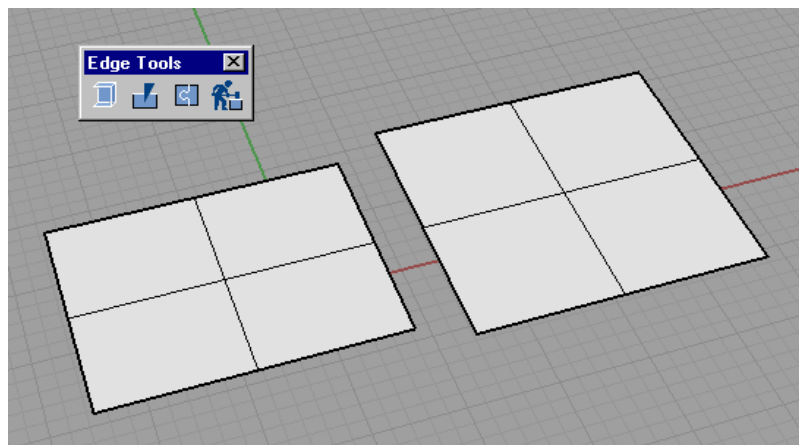
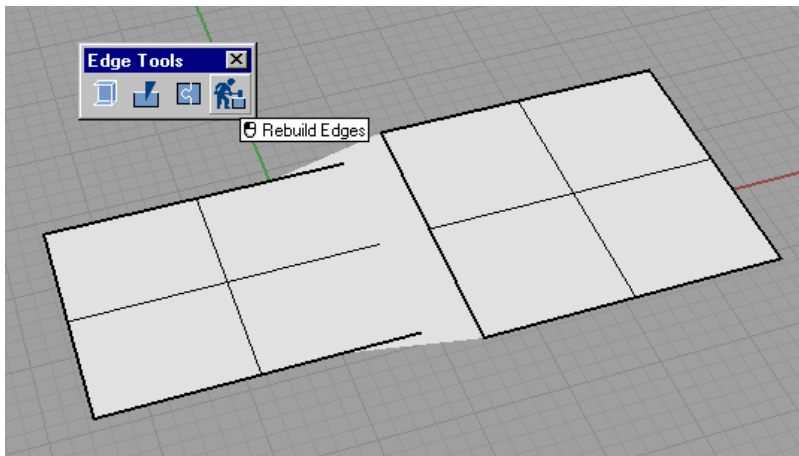


“_JoinEdges” command can be used for 3d print or CAM (when exporting meshes) but it cannot be used in any way if the file will be exported using IGES or STEP (NURBS) because gaps will appear again.

In fact using “_JoinEdges” allows Rhino to generate a mesh in the gap, which is exported later in STL format or similar mesh format for 3d Print.

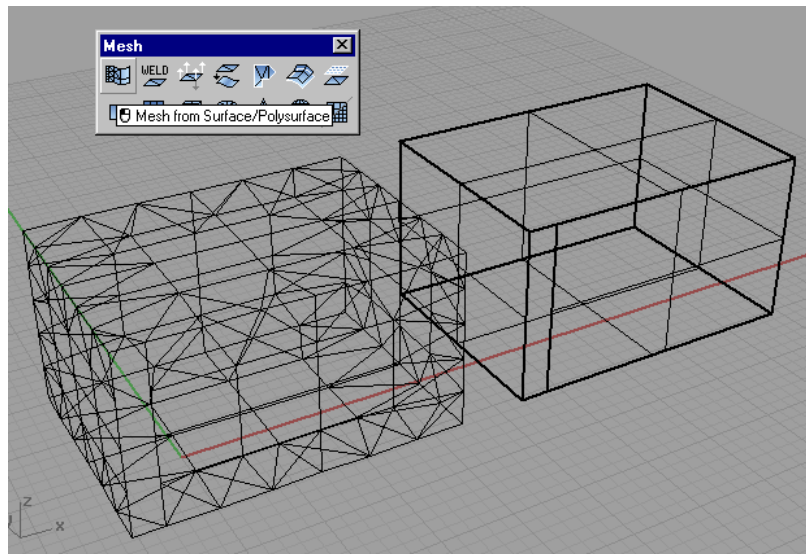
It is a quick solution, but not the best one. It could be handy in some cases, but it is always better to have a clean, closed (no naked edge) model.

To undo the “_JoinEdges” effect you can use “_RebuildEdges” command. RebuildEdges return surface edges to its original position. Before using “_RebuildEdges,” Polysurfaces must be separated using “_Explode”.



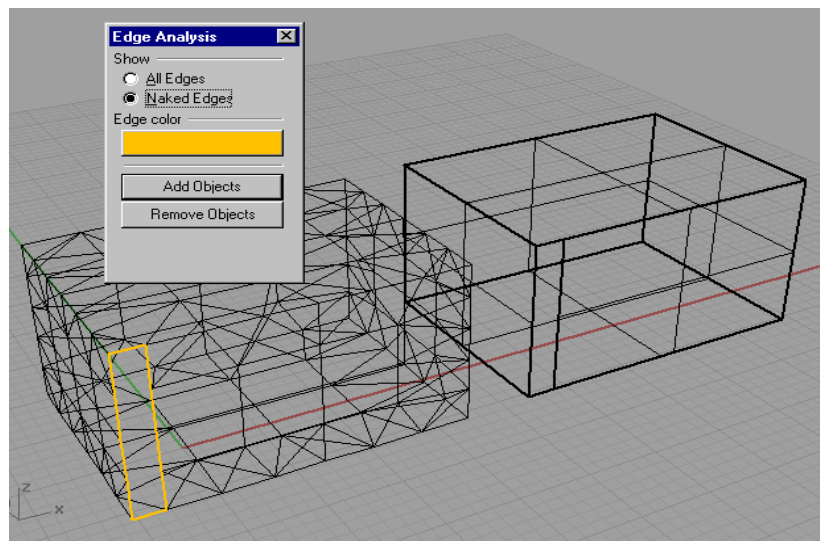
If naked edges are found, the best method to solve the problem is to identify what surfaces aren't good or are missing and recreate them to fill the gap. Create the new surface in perfect contact with the surrounding surfaces edges then use join to create a Polysurface.

There is another way to prepare NURBS models for 3dPrint. Convert NURBS models to meshes. Run “_Mesh” command and you will be able to create a mesh based on NURBS geometry (Save your NURBS model before doing this, because doing this in reverse (convert meshes to NURBS) is not simple).



Once the NURBS object is converted to meshes we must verify that the mesh has no naked edges.

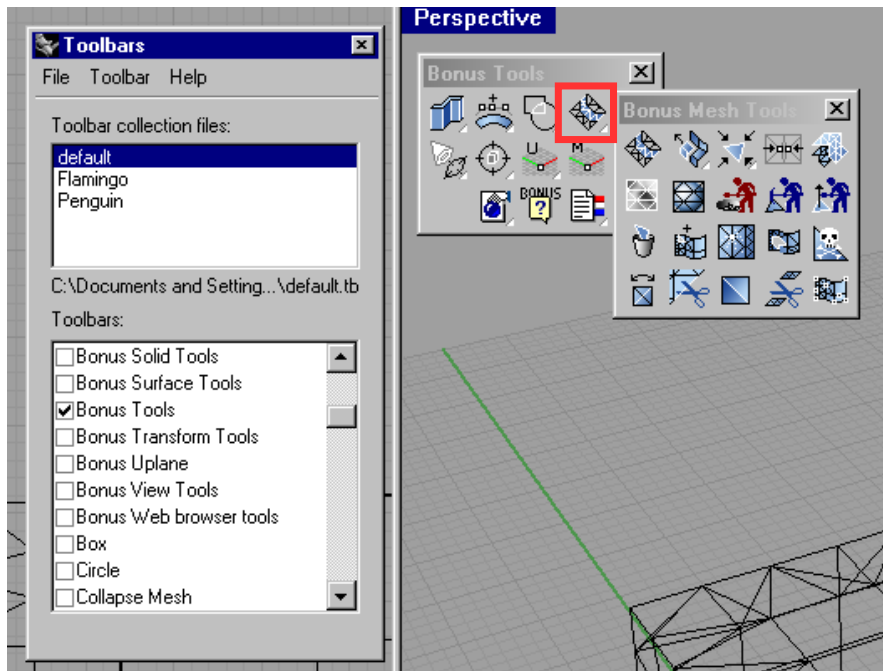
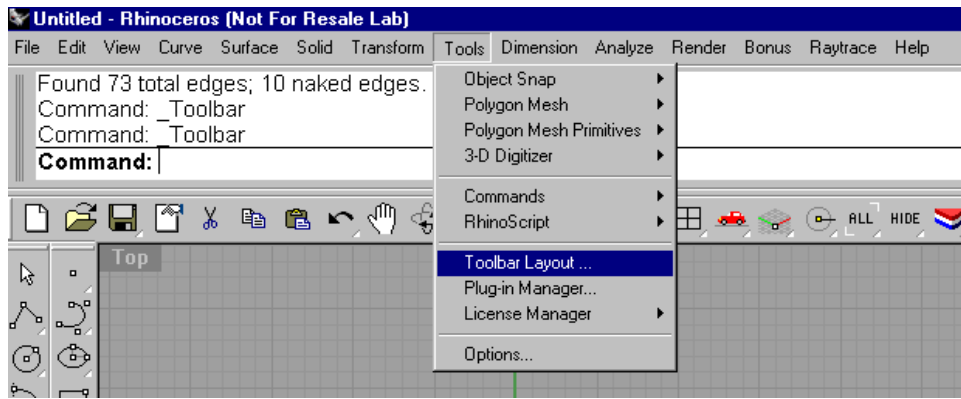
Use “_Showedges” command in the same way it is used for NURBS.



Once naked edges are detected we can close mesh gaps easily using “Fillhole” or “Fillhole” commands.

These commands belong to the Bonus Tools commands added in Rhinoceros 3.0 Service Release 4.

Look for Tools>Toolbar layout menu, select “Default” toolbar collection and check “Bonus tools” toolbar.

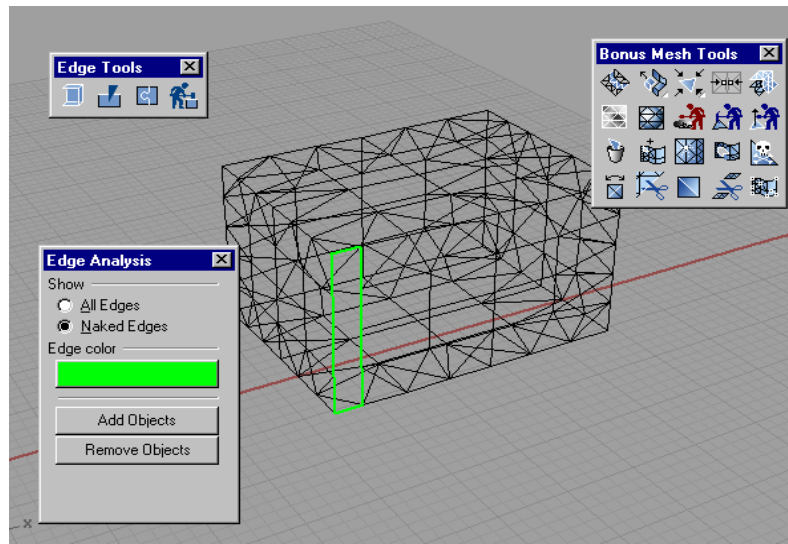


“FillHole” and “FillHoles” commands are included in Rhinoceros 3.0 Service Release 4 and later (Jun 15 2005). Registered users can download Service Releases for free.

More information: [HYPERLINK](#)

"<http://download.mcneel.com/whatsnew.asp?rel=51>". These Bonus Tools aren't available in Rhino versions previous to Rhinoceros 3.0 Service Release 4.

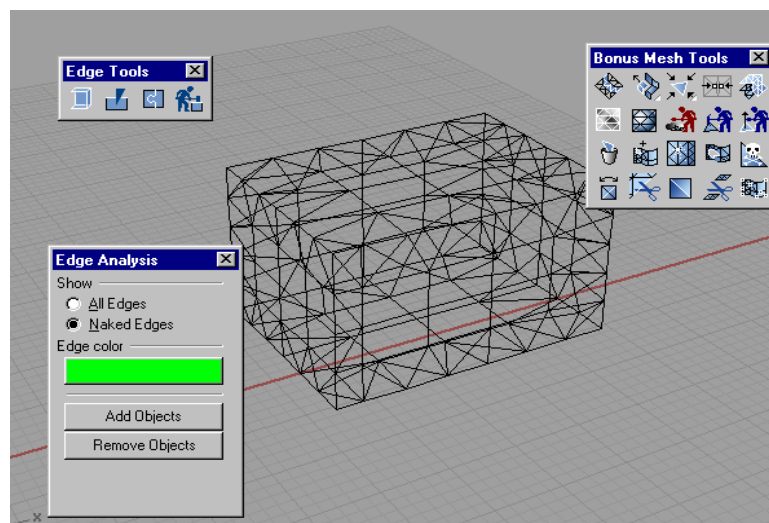
Now we will analyze the mesh model to detect where there are naked edges.



Run “_FillHole” command.



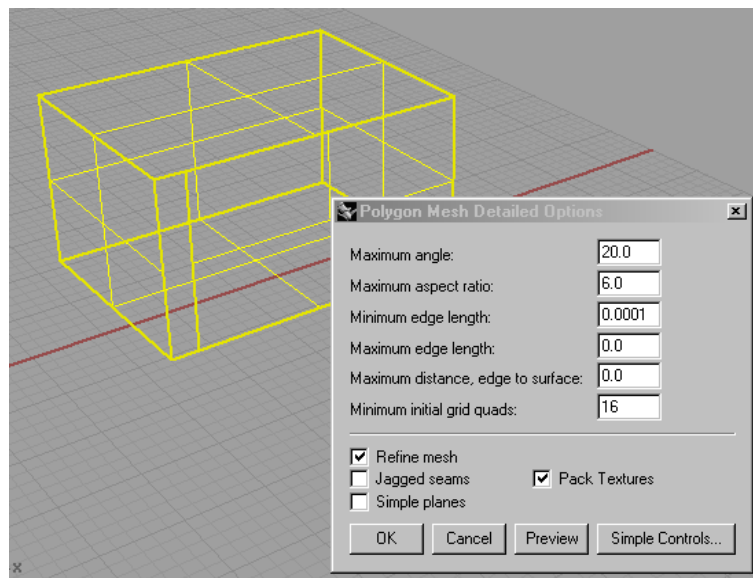
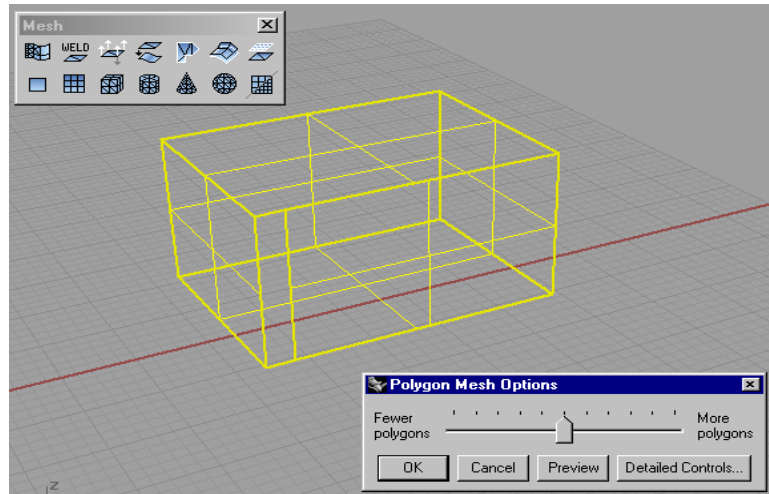
Select gap edge and it will be automatically filled with mesh polygons. No naked edges.



Converting a model NURBS into a Mesh polygon

When converting a NURBS model in Meshes a Polygon mesh options window shows up. There are 2 options, Simple Controls or Detailed Controls.

It is always better to use detailed controls rather than Simple Controls. Using Detailed Controls allows you to have more control on polygon creation.

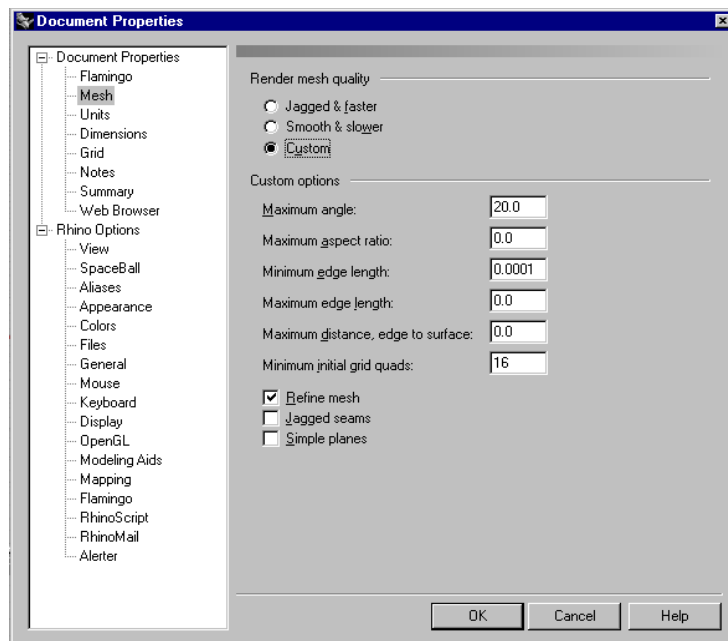


The same window shows up when exporting Nurbs models and choosing a polygon only file format. Some file formats do not support NURBS and must convert NURBS info in Polygon mesh info. Polygon formats are STL, DXF, DWG, OBJ, 3DS, RAW and some others.

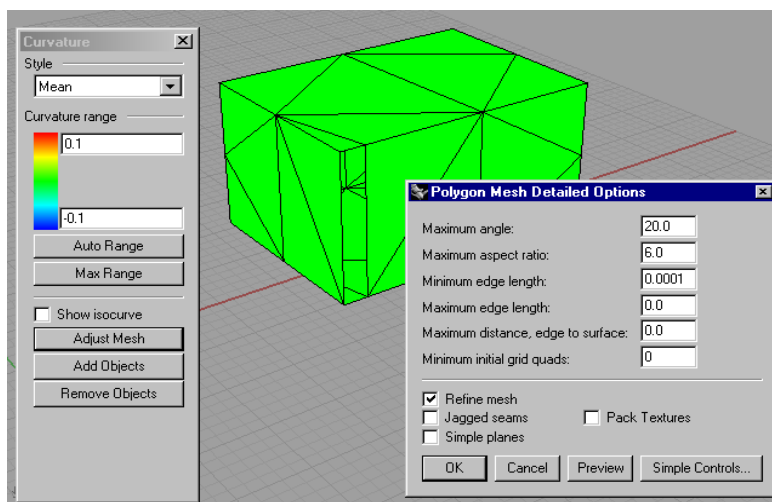
NURBS file format are 3DM (Rhino native format), IGES and STEP. There are other proprietary formats that also support NURBS.

NURBS to MESH options.

This mesh options are the same we can find in Document properties>Mesh for render mesh quality and choose “Custom” option.



There are also some same options found in surface Analysis. These options aren't linked, but the parameters used are the same ones on both.



Parameters changed on surface analysis aren't translated to render mesh parameters or export parameters or vice versa. These options are totally independent from each other.

MESH FAQ

Information obtained from HYPERLINK:

"<http://en.wiki.mcneel.com/default.aspx/McNeel/RhinoHome.html>"<http://en.wiki.mcneel.com/default.aspx/McNeel/RhinoHome.html>

Specifically from: <http://en.wiki.mcneel.com/default.aspx/McNeel/MeshFAQ.html>

Why do we need meshes anyway?

Although Rhino is a NURBS surface modeler, it uses polygon meshes created from those surfaces for visualization purposes - so what you see on the screen when the model is shaded is in reality a special, invisible polygon mesh (the "render mesh") that is attached to the actual NURBS surface.

Why? For quick shading and rendering. But while a mesh has the advantages of shading speed and adjustability, there is a disadvantage. The render mesh is always an approximation of the surface, so there are almost always gaps between the faceted render mesh and the actual smooth surface.

The same meshing engine in Rhino is used in the creation of analysis meshes for the functions like Draft Analysis and Curvature Analysis, EMap, etc., and by the **Mesh** command (Tools > Polygon Mesh > From NURBS Object), which creates a "real" mesh object directly from a NURBS object.

Exporting from Rhino with certain polygon-based formats (such as .stl) will also create mesh objects (in the exported file). Although you can't edit these in the original Rhino file, you do have the same group of mesh density settings to control how they are created.

The functions that create real, editable meshes like **Mesh** and **Export** (.stl) are very important in many applications, as other "downstream" programs or processes often need polygon mesh objects to work with.

All of the types of mesh objects can exist simultaneously in one file, they do not affect one another and each can have its own settings. Although all the meshes are created with the same engine, there are a couple of important differences between them.

The different types of meshes

The **render meshes** created on NURBS surfaces and polysurfaces for visualization purposes (by using Shaded or Rendered Viewport) are not user accessible or editable, and cannot be separated from the NURBS object they were created from. They can be deleted by using the **ClearAllMeshes** command, and regenerated by using the **RefreshShade** command or by changing the settings (at File > Properties > Mesh) (which forces a global regeneration of all render meshes).

The **analysis meshes** are similar to the render meshes in that they are not editable, and cannot be separated from their NURBS object. They simply have another set of controls and exist separately from the render meshes. You *can* see them temporarily, however, when you use the "adjust mesh" button or the "preview" button in the settings panel on the dialog boxes for the Analysis commands.

The meshes created by the **Mesh command** are visible and editable, and separate from the NURBS objects they were created from. They are objects in their own right, and can be seen and edited with the various Rhino commands that apply to meshes (see Bonus > Mesh), and can be exported to in the polygon mesh formats like STL, DXF,

3DS, and OBJ.

Meshes created during **Save and Export** (such as .STL) have the same mesh settings dialogs ("Detailed Controls") as the other types. You can also see them temporarily when you use the "preview" button in the settings panel, but they are not stored in the original file (only exported).

***Willem:** When meshing an object for export I always first create a mesh and look at it in the viewport Flat Shade mode (**FlatShade** command) before exporting it to the desired format. The Flat Shade mode "Shades the current viewport with no smoothing so the individual (render) mesh faces are visible".*

The Mesh Settings Dialogs

The controls for different mesh types are virtually identical. The controls for the Render Mesh settings are part of the .3DM file's properties, they are global for the whole model. When exporting to a mesh format like STL, DXF, 3DS, and OBJ you can alternatively use simple controls, like a slider. The detailed controls for all are identical, and are described further below.

The render mesh (display mesh) settings can be found at **File > Properties > Mesh**. Rhino offers you 2 "standard" settings, "**jagged and faster**" and "**smooth and slower**", as well as "**custom**", which lets you access the detailed controls. The "simple" controls for the others are displayed at the moment you invoke the command and are just a slider: **less <-> more** polygons. The "detailed controls" for the others are all the same.

Default settings

The default for render meshes is "**jagged and faster**", which is fine for quick visualization, but not very good for anything else.

The "**smooth and slower**" theoretically offers finer resolution at the expense of longer meshing times, but in practice, even though it does take longer, frankly, it may still leave visible gaps where you don't want them, so you are advised to try the "**custom**" settings instead.

The default "simple" slider settings for analysis meshes as well as Mesh and Export is "somewhere in the middle"...

The default setting for the special STL Export dialog is the value of Absolute Tolerance at File > Properties > Units.

The Custom Settings

If you really want to control your meshing process, here is where you need to start!

There are six numerical settings and 3 check boxes. Each one has a different method of mesh control and some of them can work together. The interactions and combined effects of these settings is complex to understand. Individually they are well described in the Help however, and reading this info carefully will give you a good idea of what each one does.

Some Quick Guidelines

Below are quick custom setting suggestions that have been known to work, they are for you to experiment with.

Personally, I use the following detailed mesh settings:

Maximum angle **0.0**

Maximum aspect ratio 0.0

Maximum edge length 0.0

Maximum distance edge to surface **

***insert scale appropriate value here. For millimeter-unit models try 0.10 to 0.01*

Minimum initial grid quads 16

Refine mesh *checked*

Jagged seams *unchecked*

Simple planes *unchecked*. The **maximum distance edge to surface** method forces Rhino to create a mesh that is no further away from the surface than the specified value, which is in current file units. It allows Rhino to put fewer polygons in lower detail areas and more polygons in higher detail areas, resulting in a more efficient mesh.

The **minimum initial grid quads** setting makes sure the flatter areas have enough polygons to look smooth.

Comments from users:

“Pascal: *I prefer to set some largish number in "Maximum angle" rather than zero (disabled), maybe 35-45 degrees. This way 'features' that fall below the max distance number will be subdivided a little. You can get away with a little bigger max distance number in some cases if you let a large angle setting take over for the smallest objects. When there is too great a discrepancy between an object and the global mesh settings, it often pays to use the Mesh command on those objects. Set the mesh to be as loose or as fine as needed for that object, then render the mesh object only, hide or delete the NURBS surface. Very small repeated details can be meshed very very loosely for instance, without impacting the quality of the image.”*

“Mitch: *I sometimes put a value of 6.0 in the "Maximum aspect ratio" setting to keep Rhino from meshing long, thin objects with long, skinny triangles, it will break them up into more shorter, smaller ones. However, this comes at the expense of bigger file sizes and sometimes significantly longer meshing times.”*

“Ricardo: *I do jewelry rapid prototyping. My default settings are 0.001 for maximum distance and 12 degrees for maximum angle. Larger angles tend to show the triangle edges on the final product. Mesh sizes range up to 30mb on very complex models. Units are millimeters”*

The main setting is the one for max distance edge to surface. If you think about it for a moment, you will understand that this value is scale (size) dependent, hence the following question:

What is a scale-appropriate value?

First, it depends on what you're going to be using your mesh setting for. For general display purposes your value can be a bit bigger (looser), as you're only visualizing the model on your screen, and less polygons mean faster meshing times and quicker display reaction when tumbling, zooming, etc.

If you are doing computer-sized objects, 0.01mm works out pretty well, for watches and jewelry, you will probably want it smaller, maybe .002, for larger objects like buildings, much larger, maybe 1 or even larger.

If you're going to be exporting your object for later processes, or are going to be doing very detailed renderings, you may need to tighten up the settings a bit, depending on where it's going and how precise the process is. For a stereo lithography part, .01mm should still be fine for example, but for a good machined part, .001mm-.002mm is probably more appropriate.

It may seem very complicated at first, but after a bit of practice, you will find a few standard settings that work for you in most situations. These settings can even then be programmed into a macro or script to quickly launch the meshing of your objects with the desired characteristics.

Known meshing problems and weaknesses

Sometimes, you may find that even with the settings guidelines above, you are still not getting good results. You may struggle with ghost surfaces, uneven shading, or triangular mesh facets that cut across empty space where they shouldn't be. Some of these problems may be caused by the Rhino mesher's reaction to certain types of geometric structures. The only way to correct them currently may be to do some reconstruction on your structures.

Things to watch out for:

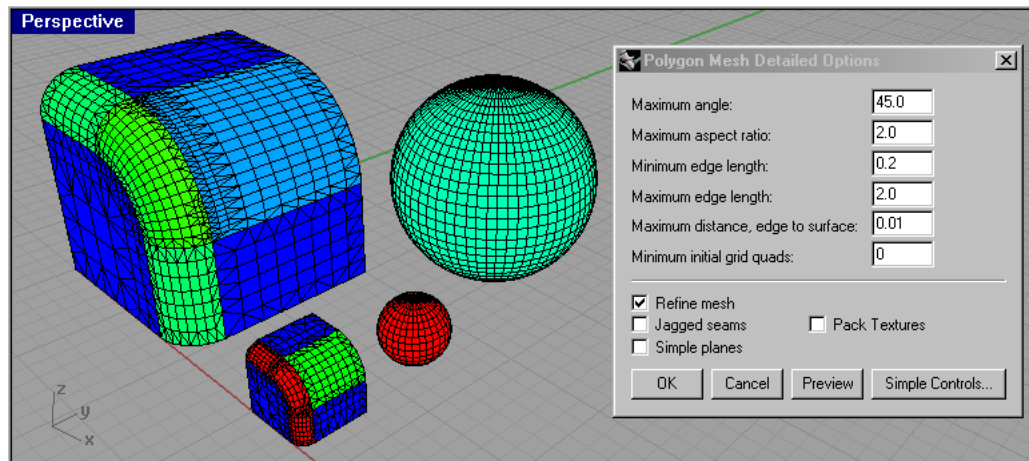
While these conditions don't always cause problems meshing, they have been known to do so in the past, so it's worthwhile checking if you are having difficulties.

Bad objects. While they don't always result in mesh problems, these are easy enough to track down, so it's a good place to start. If you find one, try hiding it, if your problem disappears, then perhaps all you need to do is fix the object (make it valid) and you'll be good to go.

Long, skinny surfaces. These are hard for the mesher currently. The longer and thinner, the harder it is and the longer it takes. A typical example might be long continuous small radius fillets on the edges of your model. If it's taking forever to mesh and you think your geometry doesn't warrant that, you might also have a tiny sliver surface somewhere that's hanging up the mesh machine.

Joined tangent lines and arcs that have been extruded or revolved. Example - extruding a rounded rectangle. This forms a single surface with internal G1 areas, which the mesher struggles with. Solution - **Explode** the curves before extruding, or use **Split > Isocurves** at the G1 spots to create a structure with separate joined tangent surfaces instead of one single one. In the case of the extruded rounded rectangle, you will have eight joined surfaces, not one.

Kinked Surfaces. Usually caused by using **MergeSrf Smooth=No** on surfaces that are not at least tangent to each other, or point editing of some types of surfaces under certain conditions. In this case, again, it is better to have multiple joined surfaces instead of one kinked one. Use **Surface edit tools > Divide surface along creases** or **Split > Isocurves** at the kinked spots to split these types of surfaces up into separate parts.



Mesh Creation Detailed Controls.

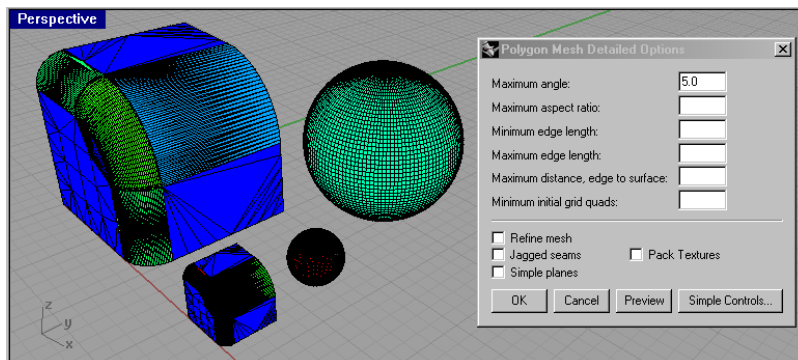
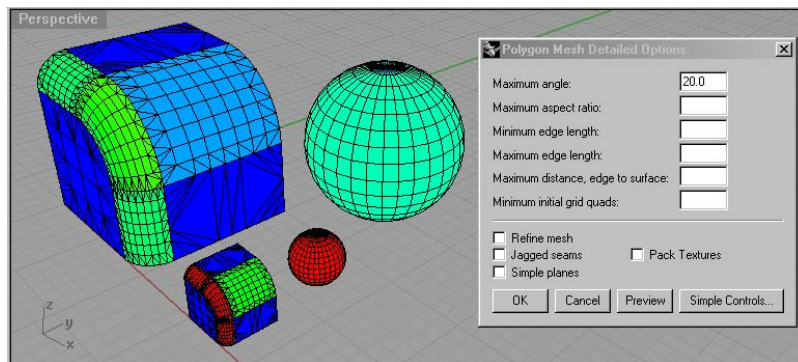
Maximum angle

Smaller values result in slower meshing, more accurate meshes, and higher polygon count. The Maximum angle is the maximum allowable change between the surface normal at any point and the mesh vertex.

Geek Speak: The maximum angle between surface normals at neighboring vertices is less than or equal to Maximum angle. Two vertices are neighbors if they are at the ends of a facet edge.

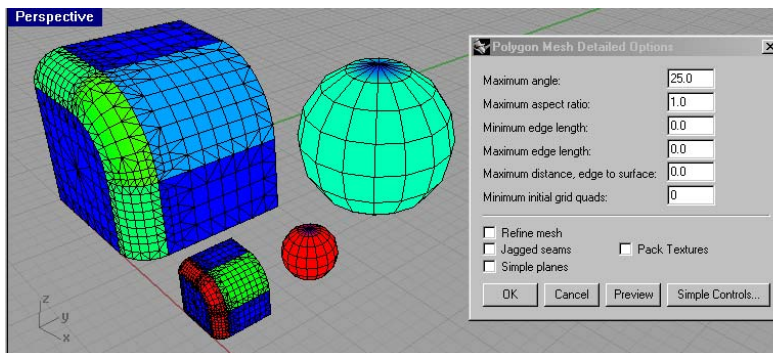
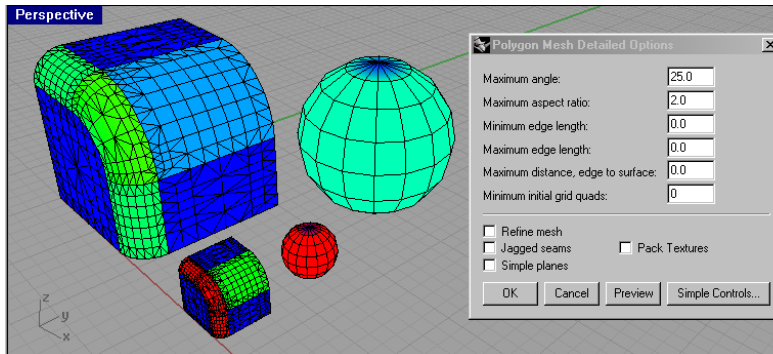
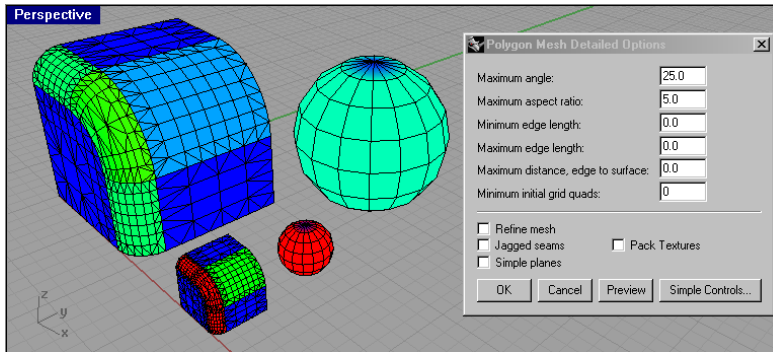
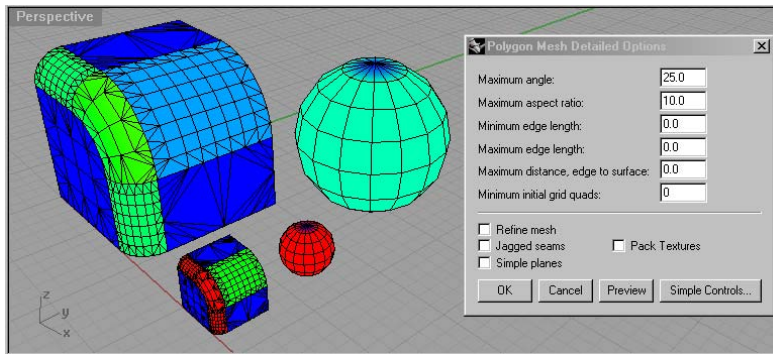
What it means to you: When you are meshing surfaces that have flat and wiggly regions and you want fewer triangles in flat regions and more triangles in wiggly regions, then use max angle and set maximum edge length=0.

Experiment Make a model with a sphere, torus, and cylinder. Then scale these objects to make a small, medium, and large-sized version. Now vary Maximum angle and notice the results.



Maximum aspect ratio

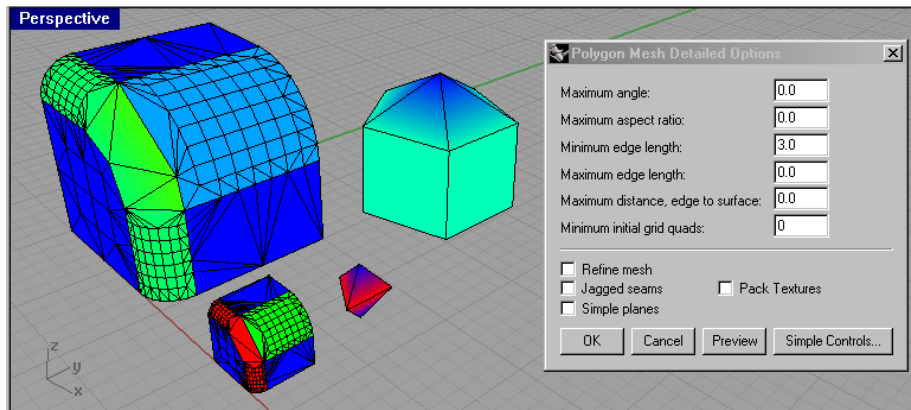
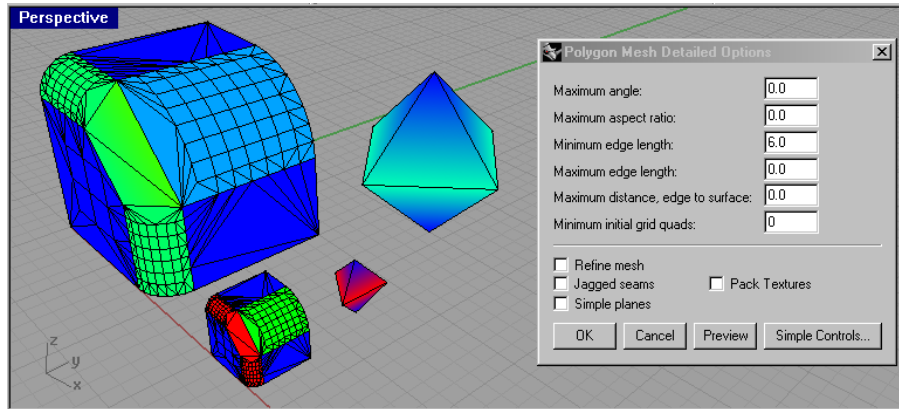
Surfaces are initially tessellated with a regular quadrangle mesh and then that mesh is refined. The initial quad mesh is constructed so that on average, the maximum aspect ratio of the quads is less than or equal to Maximum aspect ratio. Smaller values result in slower meshing and higher polygon count with more equilateral polygons. This is approximately the maximum aspect ratio of the quads in the initial mesh grid. Setting Maximum aspect ratio to zero turns off the option. Zero means no limit. The default value is 6 and the suggested range is from 1 to 100. Scale independent.



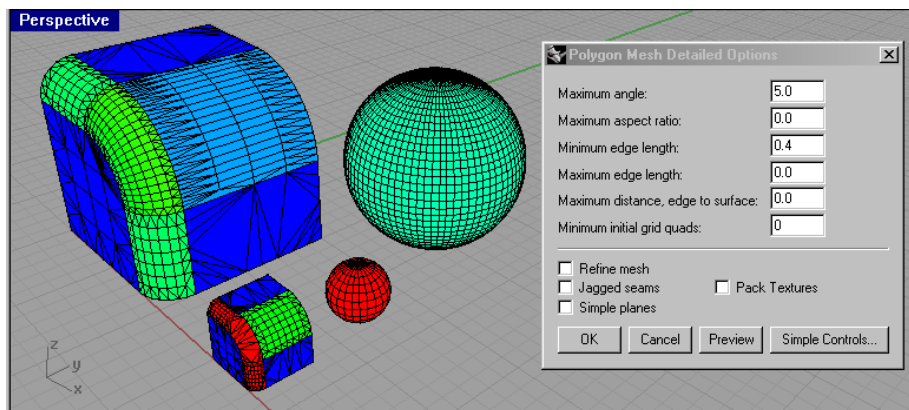
Minimum edge length

The value is always in the current unit system.

Bigger values result in faster meshing, less accurate meshes and lower polygon count. Setting Minimum edge length to zero turns off the option. Limits the size of the polygon edges. Default is 0.01 and the usable range depends on the size of the model. Scale-dependent.



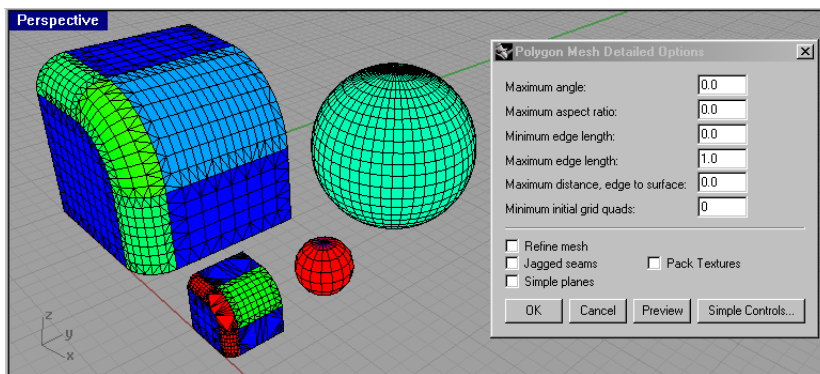
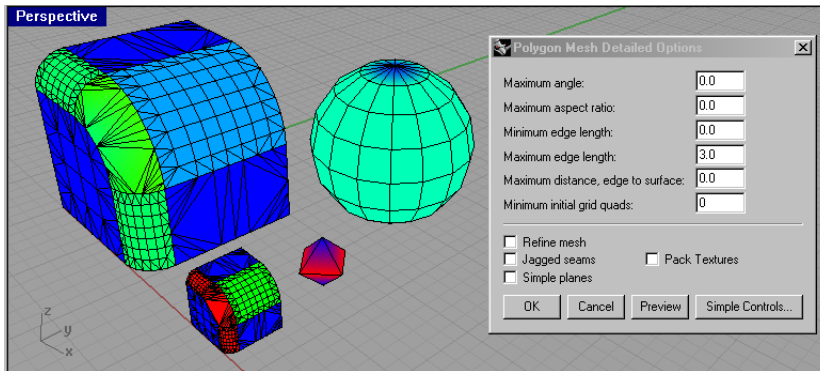
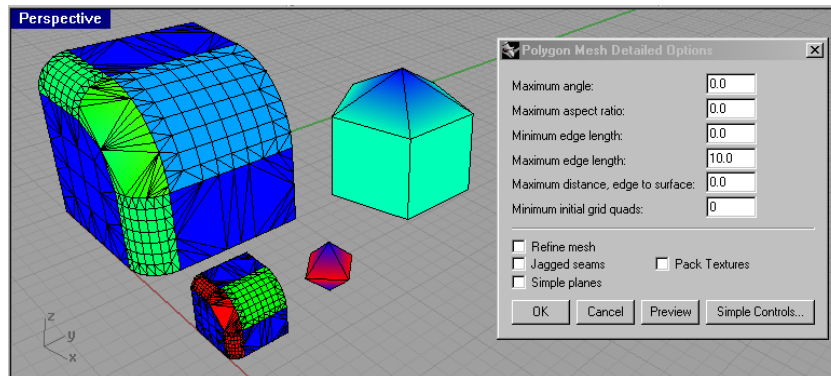
Combined with maximum angle is used to prevent too small polygons creation.

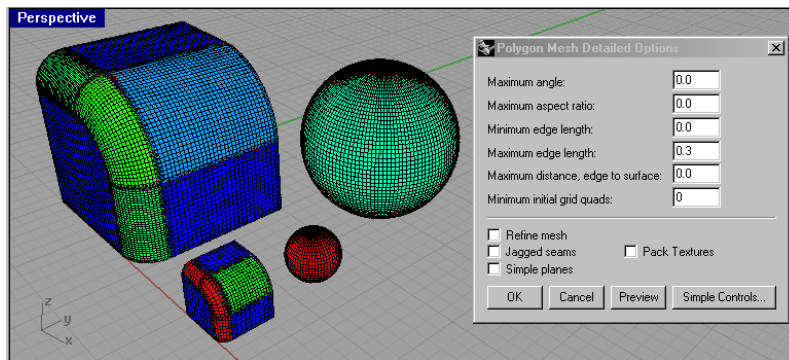


Maximum edge length

The maximum length of an edge will be less than or equal to Maximum edge length.

Smaller values result in slower meshing and higher polygon count with more equally sized polygons. Setting Maximum edge length to zero turns off the option, which is the default. When the Refine is checked, polygons are refined until all polygon edges are shorter than this value. This is also approximately the maximum edge length of the quads in the initial mesh grid. This option is scale dependent and can be used for making sure the polygons are approximately the same size.

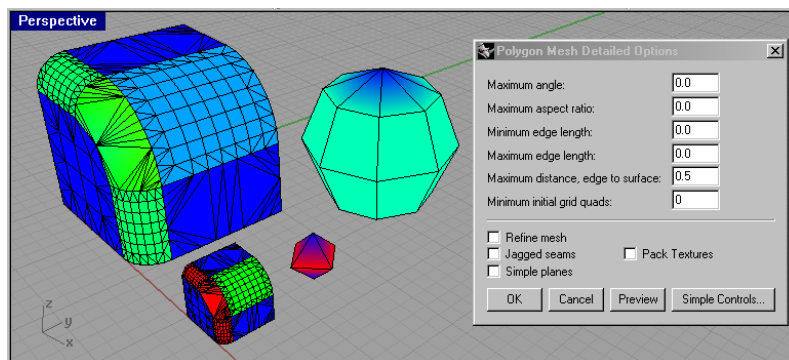
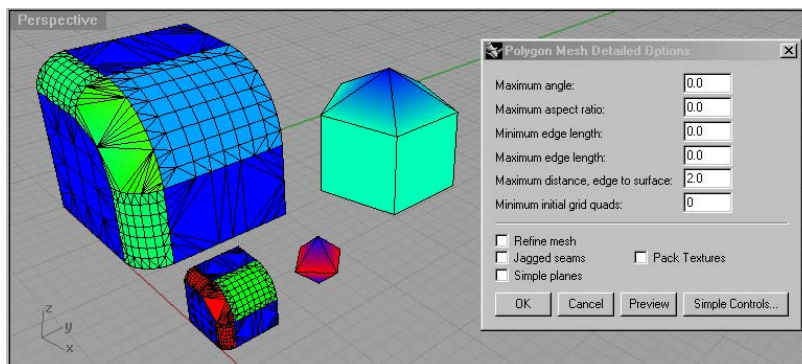


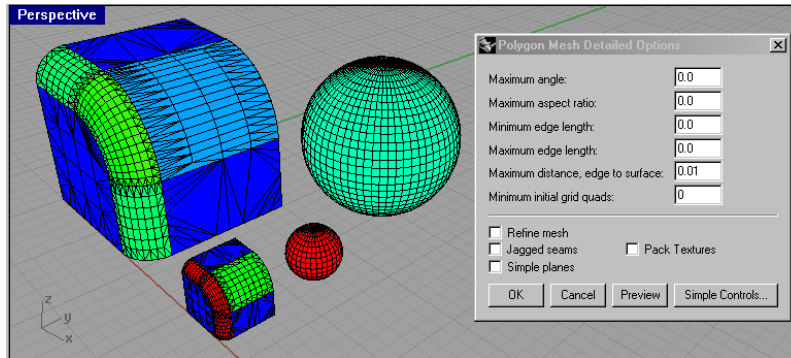


Maximum distance, edge to surface

The maximum distance from the center of an edge to the surface will be less than or equal to Maximum distance, edge to surface.

Smaller values result in slower meshing, more accurate meshes, and higher polygon count. Setting Maximum distance, edge to surface to zero turns off the option. The default value is zero and the usable range depends on the size of the model. When the Refine is checked, polygons are refined until the distance from a polygon edge midpoint to the NURBS surface is smaller than this value. This is also approximately the maximum distance from polygon edge midpoints to the NURBS surface in the initial mesh grid. This option is scale dependent and can be used as a general polygon mesh tolerance setting.

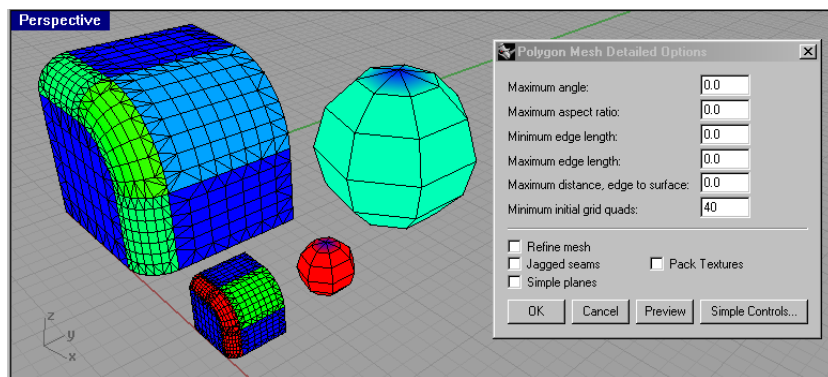
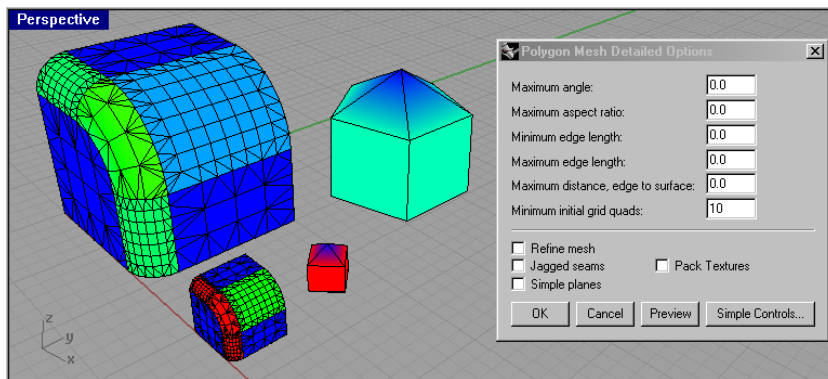


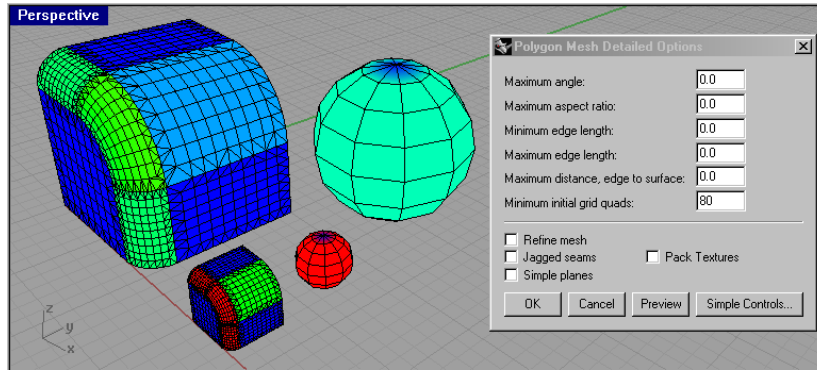


Minimum initial grid quads

Minimum number of quads in the initial mesh.

Bigger values result in slower meshing, more accurate meshes and higher polygon count with more evenly distributed polygons. Setting Minimum initial grid quads to zero, turns off the option, which is the default. Forces simple geometry to be meshed with more polygons. Zero means no minimum. The default is zero. Scale independent. This is the number of quads per surface in the initial mesh grid. In practice, Rhino will use at least this many polygons for each surface. This option is scale independent and can be used for making sure that surfaces with very subtle details are meshed with high enough polygon count.



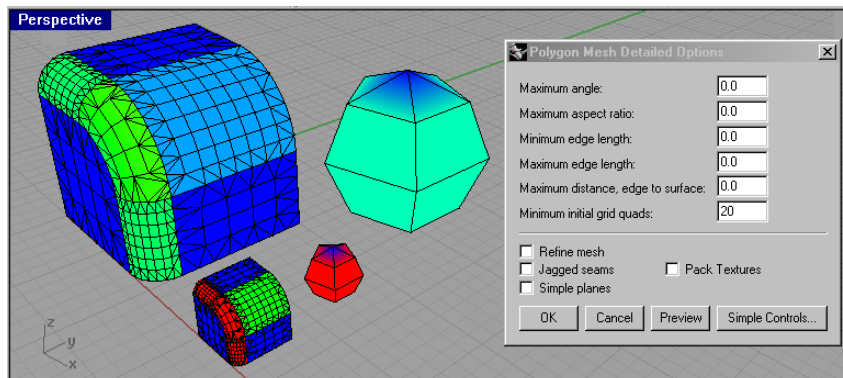
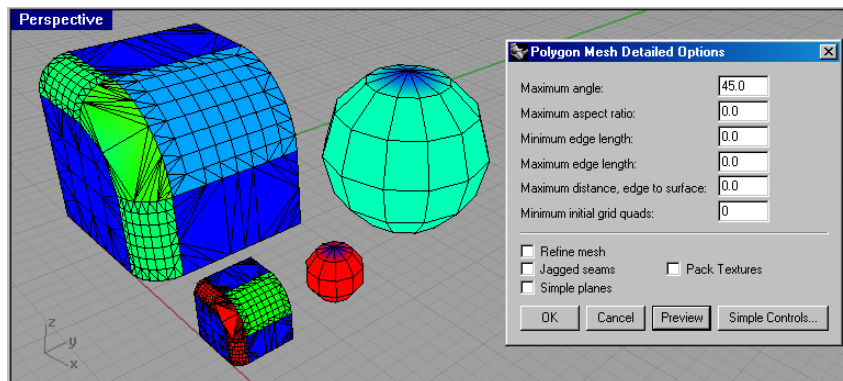


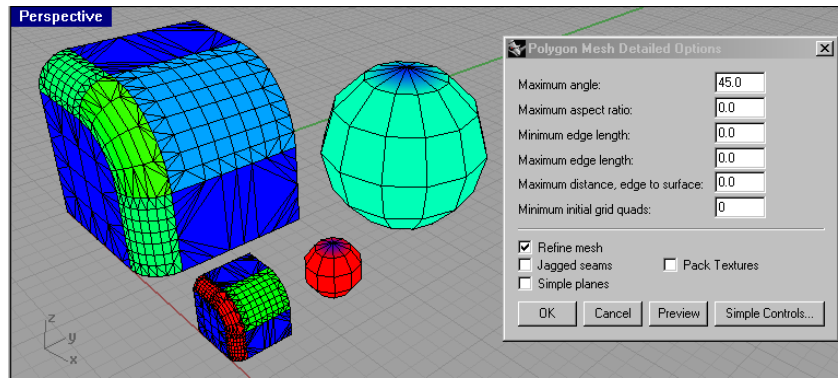
Refine mesh

If refine is not checked, the initial quadrangle mesh is returned.

The mesh is refined until the angle between surface normals along a polygon edge is smaller than this value. The default is 20 degrees and the suggested range is from 5 to 90 degrees. Setting Max angle to zero turns off the option. Scale independent.

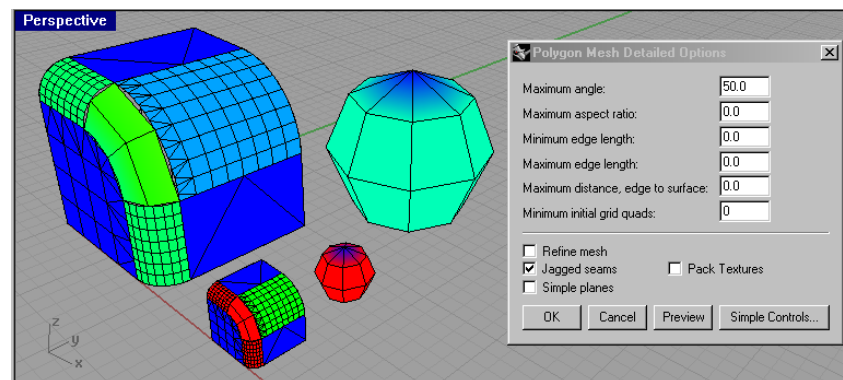
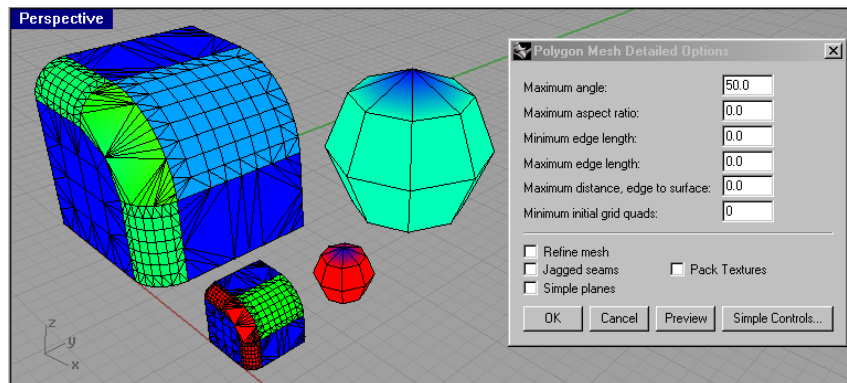
No refinement results in faster meshing, less accurate meshes, and lower polygon count. Clearing this check box also means untrimmed individual surfaces and surface areas away from trim edges and joined edges are meshed with evenly sized quadrangles. When this Refine is checked, after initial meshing, Rhino uses a recursive process to refine the mesh until it meets the criteria defined by Maximum Angle, Minimum edge length and Maximum edge length and Maximum distance, edge to Surface options.





Jagged seams

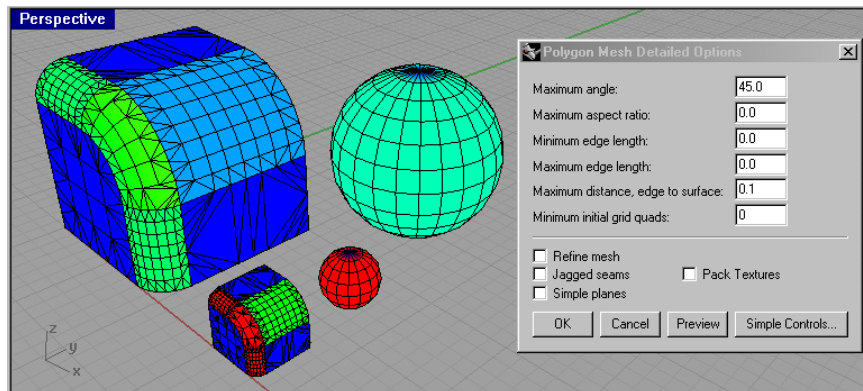
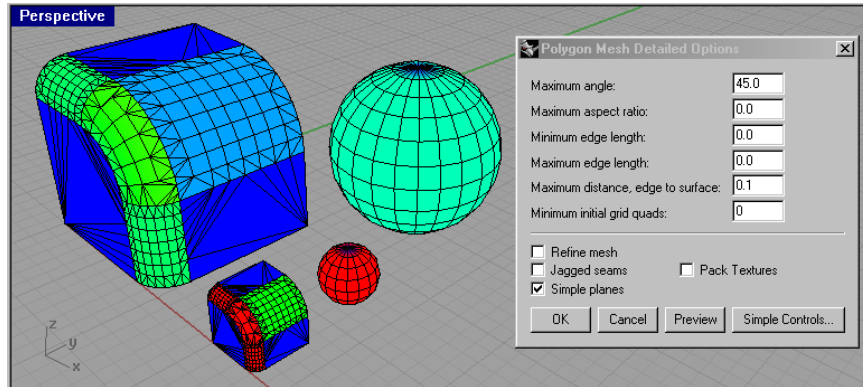
If checked then meshes for each surface in a polysurface do not necessary meet to form a watertight mesh. If not checked, then water tight meshes are created. Causes dramatically faster meshing, lower polygon count and cracks between joined surfaces in the rendered image. By default, Jagged seams is not selected. This options means that all surfaces are meshed independently and the meshes of joined surface edges are not stitched together.



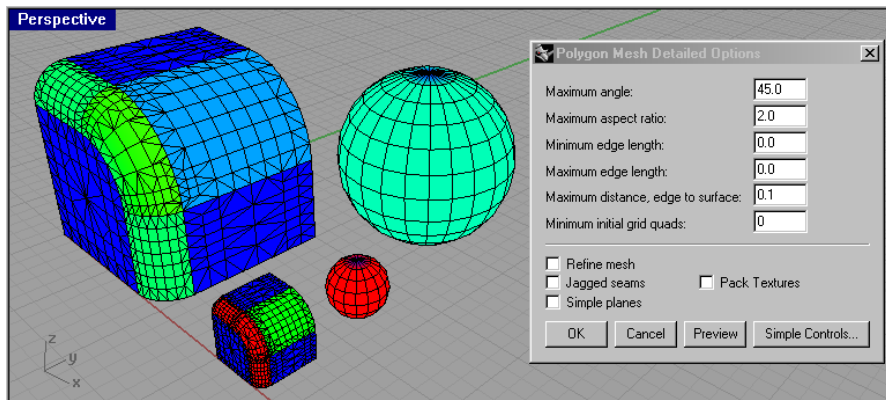
Simple planes

If simple planes is checked, then the above settings, except Jagged seams, are ignored for planar surfaces and the planar surface is meshed with as few polygons as possible.

Causes slower meshing and minimum polygon count on planar surfaces. Meshing can be slower especially for complex trimmed surfaces. By default, Simple planes is not selected. This option means all planar surfaces are meshed by meshing the surface edges and then filling the area bounded by the edges with triangles.



Mixed with maximum aspect ratio



Weld

Welds coincident mesh vertices that came from seams between tangent surfaces of a polysurface.

In general, it is the target application and not the file format that matters. In general, Weld should not be checked; especially if you are exporting to an application that cares about texture coordinates or accurate vertex normals. If you are exporting to an application that has trouble with coincident vertices then check Weld. For simple formats like STL (which has no vertex normals and no texture coordinates), welding does nothing.

Pack Textures

When polysurfaces are meshed, the packed texture coordinates are created. A packed texture is a partition of the unit square into disjoint sub-rectangles so that one bitmap can be used to apply independent textures to each face of the polysurface.

The Pack Textures option will pack the mesh textures of every mesh and render mesh in the selection set. If more than one object is selected, the packing "spreads" the texture over all the objects.

Preview

Previews the mesh that results from the current options.

Simple Controls

The Polygon Mesh Options or STL Mesh Export Options dialog box appropriate for the action appears. These simple controls are an alternate way to control the way the mesh is made.

NoteThe meshes created by the Mesh command are visible and editable, and separate from the NURBS objects they were created from. The meshes created by Rendered Viewport and Shaded Viewport on NURBS surfaces and polysurfaces are invisible, not editable, and cannot be separated from the NURBS object, except to destroy them with the **RefreshShade** command. Render meshes are controlled by a different set of meshing settings, which are on the Document Properties dialog box Render page.

Accuracy (<http://www.rhino3d.com/accuracy.htm>)

Since many free-form modelers are not accurate enough for manufacturing or engineering analysis, and since Rhino is a free-form modeler, many people assume Rhino is not accurate enough for their application.

In fact, Rhino is **just as** or **even more precise** than most CAD software.

Here are the details:

There are two common methods 3-D models are stored in computers.

The first method is using meshes (sometimes called facets), which are usually used for rendering, animation, or conceptual design. While mesh modelers often have what appear to

be precise techniques for creating models like spheres, boxes, splines, or even NURBS, behind the scenes they eventually turn everything into a mesh. Meshes are inherently inaccurate because a mesh is simply a collection of flat triangles. Even if the surface is curved, a mesh modeler still represents it with flat triangles. This is fine for most renderings, animations, and games, but not when designing for manufacturing. It should be noted that many manufacturing processes use meshes but the mesh density must be under the control of the manufacturing application to achieve the desired accuracy. Rhino does not use meshes for modeling, but it can convert NURBS to meshes at any density as needed for file exports and rendering.

The second method is [NURBS](#). Most CAD, CAM, CAE, and CAID modelers, including Rhino, represent free-form shapes as NURBS. Products that use NURBS can potentially represent free-form shapes accurately enough for the most demanding application if they are diligent in their NURBS implementation. If an application's primary focus is machinery design and not free-form shapes, it is likely that its NURBS implementation can be less than robust for demanding free-form modeling. This is typical of the mid-range feature-based parametric solid modelers that are so popular today.

Since Rhino's focus is free-form NURBS modeling, its NURBS implementation is one of the most robust available today. Here are the primary considerations when evaluating whether a modeler is accurate enough for your application:

- **Position.** Rhino, like most CAD products, represents position in double-precision floating-point numbers. That means the x, y, or z coordinate of any point can have a value ranging from as large as $\pm 10^{308}$ to as small as $\pm 10^{-308}$. Most CAD software, including Rhino, uses double-precision floating-point arithmetic.

Because of the limitation of current computer technology, we expect calculations to be accurate to 15 digits of precision in a range from $\pm 10^{20}$ to $\pm 10^{-20}$. This limitation is found in all modern CAD products.

Older CAD products often have additional limitations because they were developed originally to run on computers with less precision. For example, many CAD modelers are designed for performing calculations on geometry that is restricted to be in a box of size 1000x1000x1000 meters centered at the origin. (Geek alert: Another of the popular off-the-shelf modeling kernels requires parameterizations that are within a factor of 10 of being arc-length parameterizations.) Rhino has none of the limitations found in these older products.

- **Intersections.** In Rhino, when two free-form surfaces are intersected, the resulting intersection curve is calculated to the accuracy specified by the user. The Rhino default accuracy (tolerance) is 1/100 millimeter. Many CAD systems have built-in tolerances that the user cannot override.

If you carefully examine the geometry other modelers produce from free-form surface intersections, free-form fillet creation, and free-form surface offsets, you will discover that this free-form geometry is actually calculated with accuracy between 10^{-2} and 10^{-4} meters even though they advertise precision of 10^{-8} (without mentioning that the units are meters).

- **Continuity** (curvature change matched across a seam.) Most CAD products don't even have tools to match curvature, let alone do it accurately enough for a discriminating designer. If your application requires smooth free-form surfaces such as airfoils, hydrofoils, lenses, or reflective surfaces, you need these tools found only in Rhino or high-end surface modeling products like CATIA and Alias.

Other things to consider:

- **Units.** In Rhino the user can specify the units. The units are actually changed and then all calculations are done in those units. In many CAD products, units are only a display attribute. Even though you may have specified millimeters, all of the calculations are actually being done in meters. No big deal. You just move the decimal place over. Wrong! Read on.
- **Changing units.** Changing units or unit conversions can be one of most commonly overlooked accuracy hazard in CAD/CAM. Most of us might think that converting from imperial units to metric units would introduce some inaccuracy while never giving millimeter to centimeter conversions a thought. Why? Because we think in decimal. But guess what! The computer doesn't. It is binary (that is base 2, not base 10). That means one or more floating-point multiplies or divides are needed to convert from millimeters to centimeters. The inaccuracies introduced by converting from millimeters to centimeters are the same as those introduced by converting from millimeters to inches.

In summary, Rhino is as accurate as or more accurate than any other CAD product on the market today. In addition, Rhino provides tools for setting accuracy and units as well as tools for controlling and evaluating continuity not found in most CAD products. Rhino does not have the limitations found any of the older CAD software.

Here you can find other interesting links:

To fix IGES files that you import to Rhino, go to:

<http://www.es.rhino3d.com/tutorials/troubleshootiges.pdf>

To fix STL files that you import to Rhino, go to:

<http://download.mcneel.com/download.asp?id=stlrepair&language=es>

Author: Facundo Miri

www.la.rhino3d.com