

Polygon Reduction Technique Using Two Tools

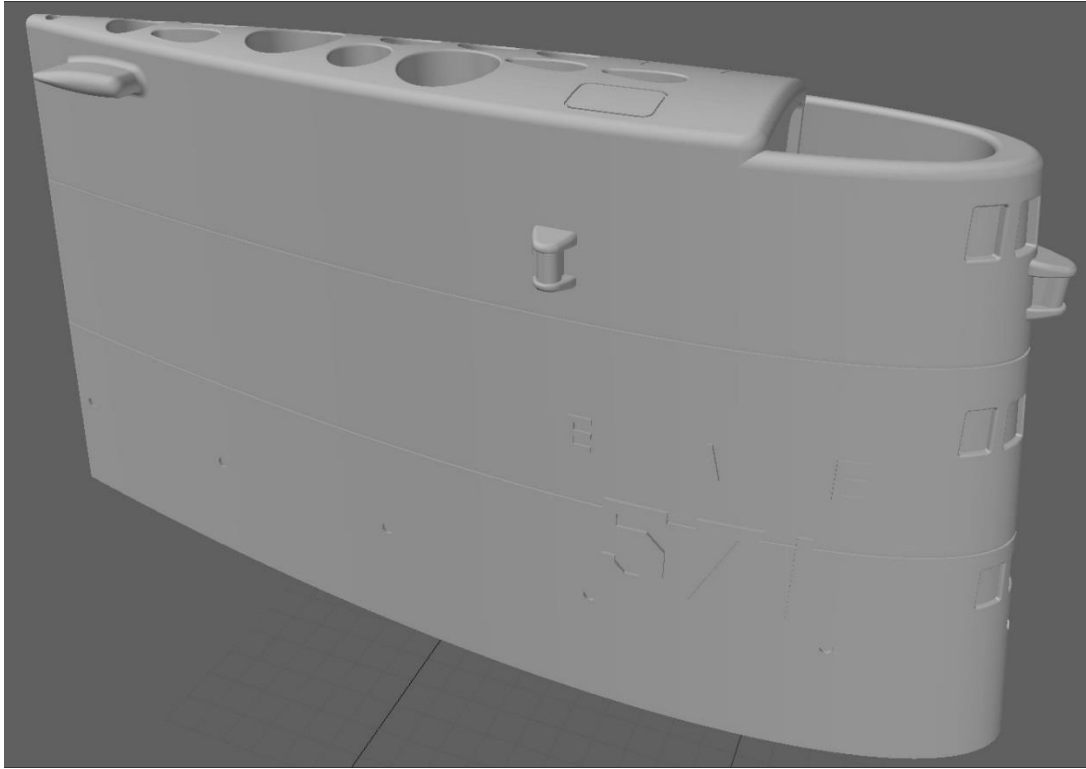
When the subject of polygons comes up, less is often more. As long as the polygons that make up an object's mesh create the desired shapes without visible faceting, better is the enemy of good enough. When building or modifying a mesh, (the wire frame result of polygonal geometry) working with the simplest mesh possible reduces the time required to get the job done. Simplicity is an integral part of a good modeling workflow.

I've been converting SolidWorks and ProE CAD meshes from design engineers into photo-realistic renders used in presentations for years using a combination of techniques to reduce the polygon count while maintaining the required level of detail. The imported files are usually in OBJ format - indistinguishable to STL (used by 3D printers) where the geometry is concerned. Conversion is tedious and time-consuming, but well worth the result – especially if the design changes, a common occurrence during development. In this tutorial, I'll demonstrate one of my go-to techniques that can be easily adapted to just about any dedicated 3D modeling software.

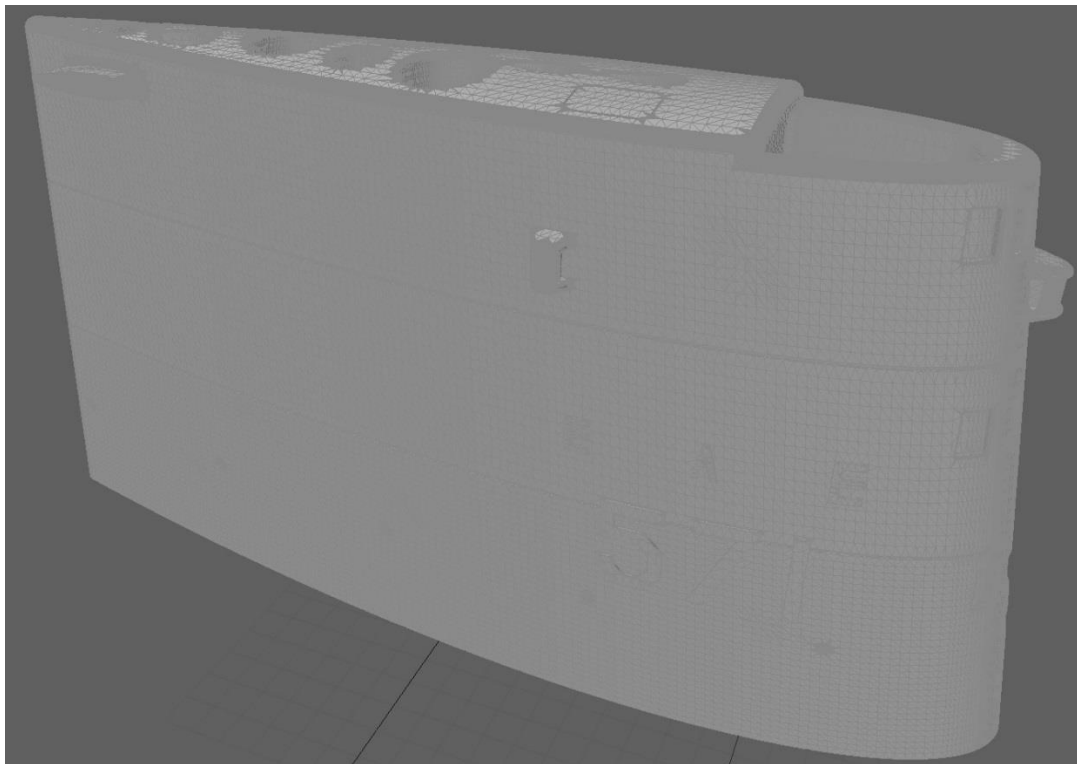
The Nautilus sail available from CGTrader has many areas with excessive polygons ripe for combining to lower the count. The total number of polygons in the sail is over 1,000,000 which is insane, but typical of CAD-generated STL files, which are more concerned with form over function.

Using the CAD-generated Nautilus sail for this tutorial, the exterior of the sail is for the most part flat, (vertically) with simple arcs to shape the bottom and top edges. This tutorial focuses on an area that was easily fixed using just two tools that have different names residing in most modeling programs. In mine, they're called Weld and Reduce Polys.

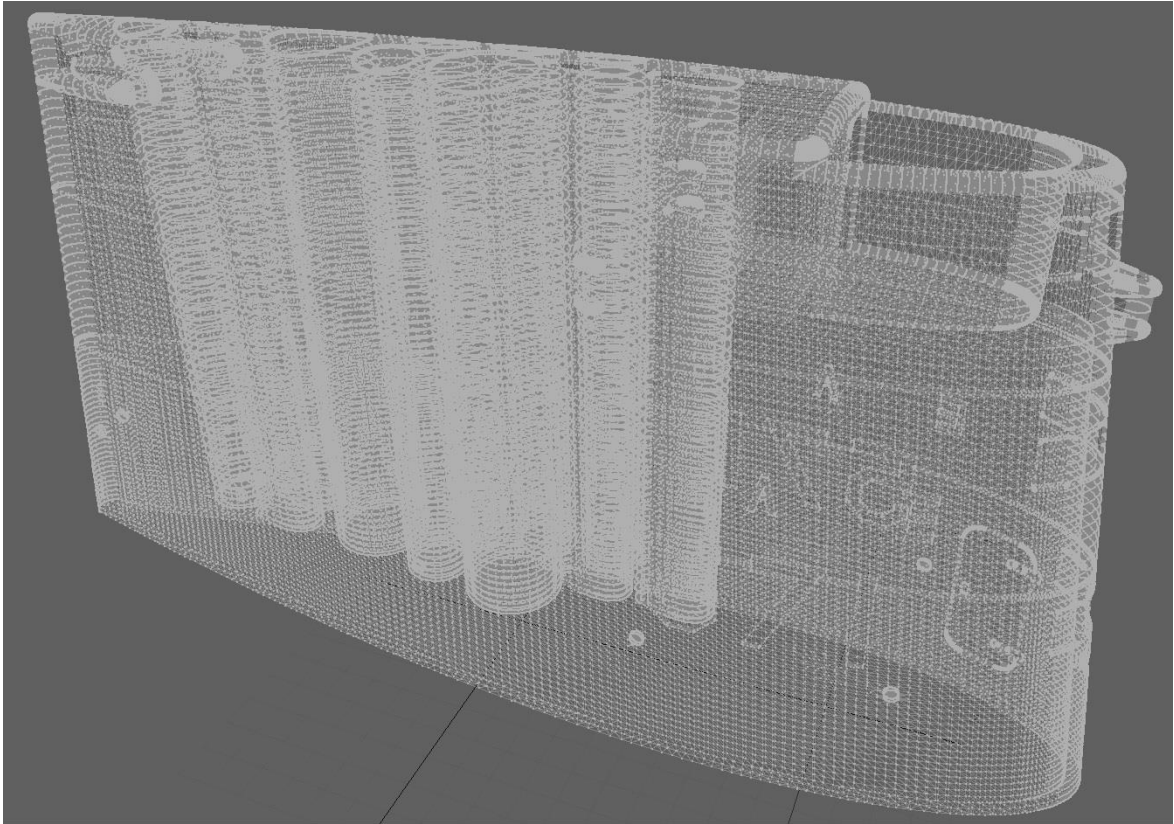
This is what a million polygon sail looks like in three different viewing modes:



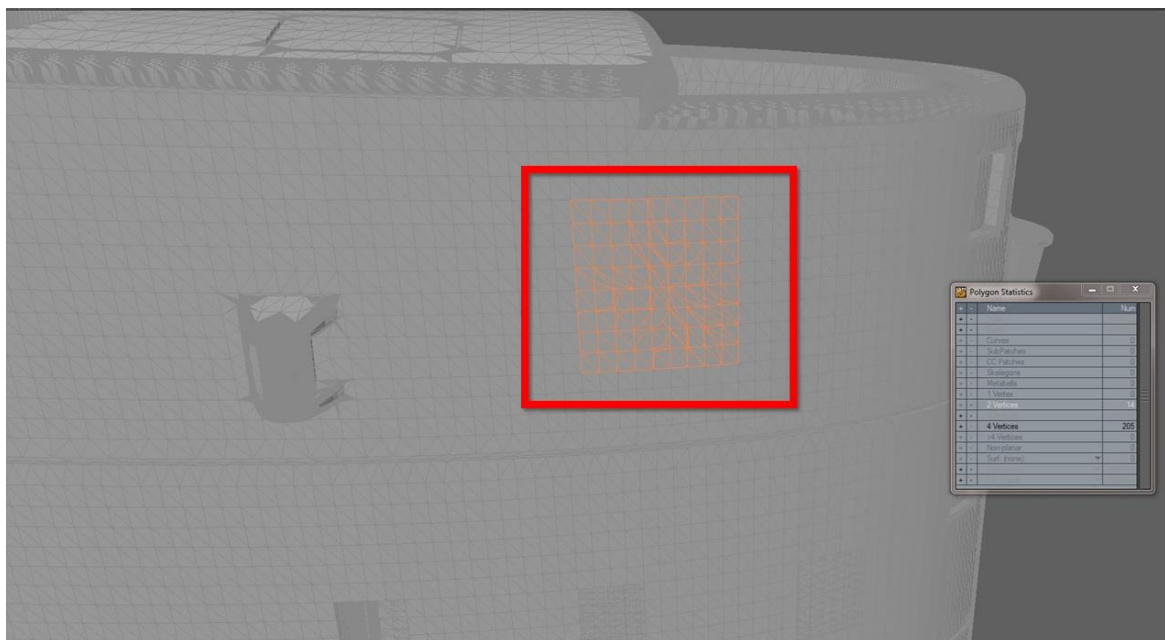
(Texture Viewing Mode)



(Textured Wire Viewing Mode)



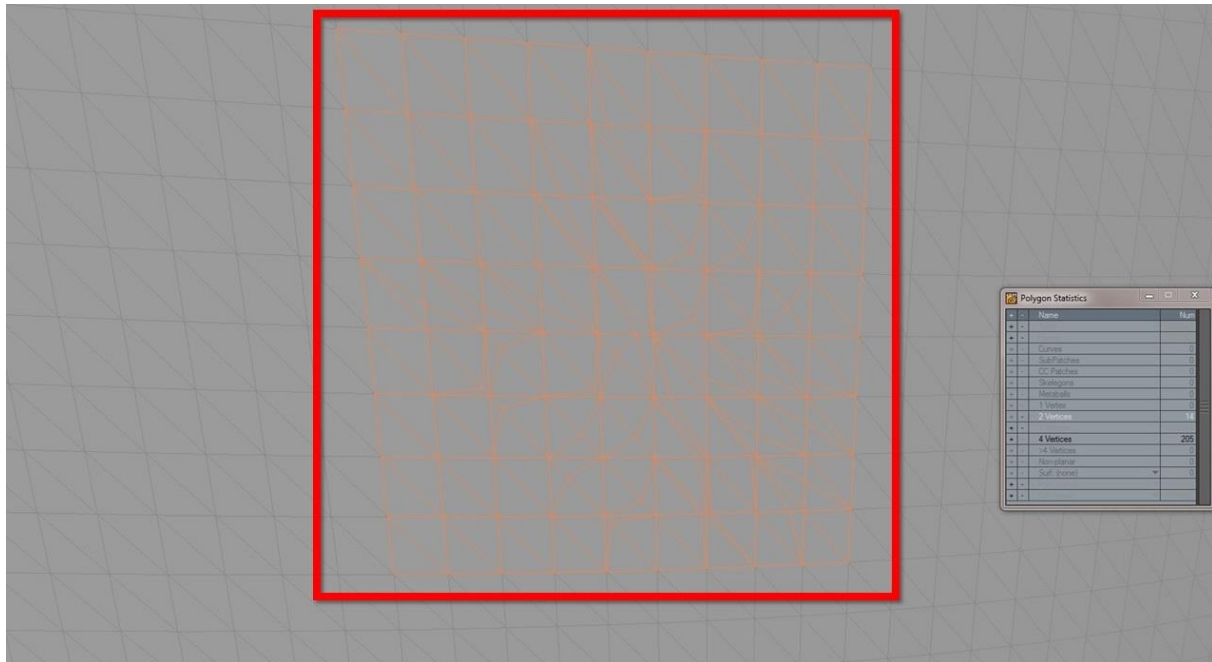
(Wireframe Viewing Mode)



(Figure 1. Area to be Cleaned Up)

Figure 1 shows the area to be polished, -just below the bridge on the starboard side.

For the eagle-eyed viewers following along, note the majority of the polygons are squares split in half, forming triangles, but inside the red square, some squares are populated with multiple triangles. The goal is to create quad, or four-point polygons. (Triangles being three-point polygons.)

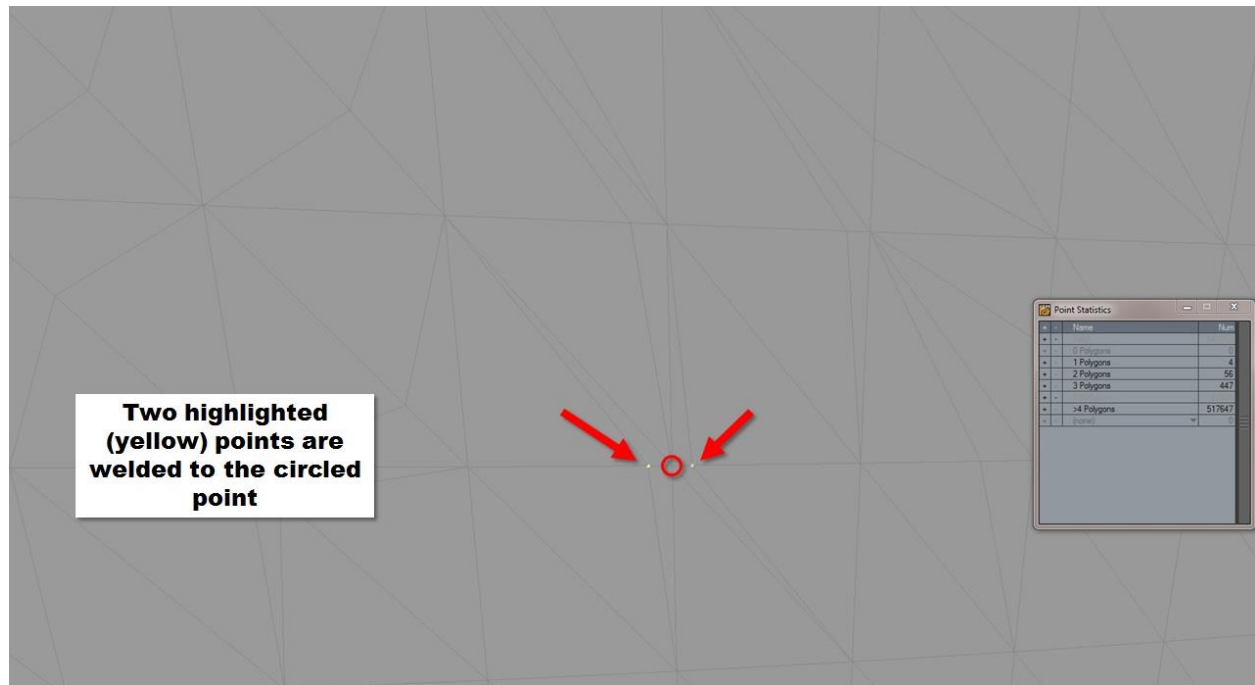


(Figure 2. Close-up of the Clean-up Area)

Figure 2 is a close-up of the same area for clarity. What could be quads are subdivided into multiple triangles, even though the overall area is flat. This is the area to be cleaned up and simplified.

Note: The excessive polygon count and resulting convoluted mesh is not the fault of the model creator. CAD programs create triangle-based meshes. The resulting high numbers of polygons occur when a model file is converted into an STL file by a CAD program algorithm that smooths out object --based on the level of detail, coarseness, or refinement selected. Higher levels of detail increase the number of polygons. (Too many polygons created during conversion can crash an entire server network as I once found out at work . . . Sorry IT twidgets!)

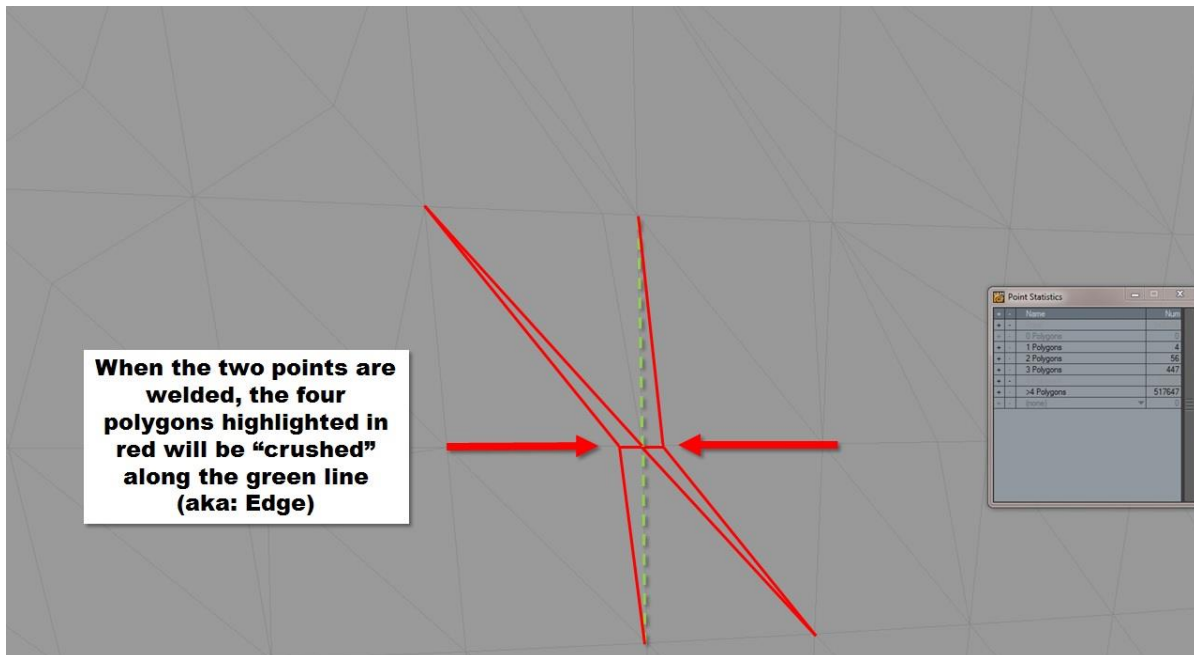
Every STL mesh is made of triangles too. In dedicated 3D modeling programs, quad poly meshes are always desirable. They're easier to work with and modify, especially when modeling complex, organic shapes. When an all-quad object is converted to an STL file, it will always be transformed into triangles, doubling the polygon count at a minimum.



(Figure 3. Three Points Welded into One.)

Figure 3 illustrates the first step. Using the polygon “flow” made up of two triangle-based, rectangular-shaped polygons, the major intersecting lines that flow vertically and horizontally across the exterior of the sail in the preceding two figures make up the two X and Y axis references needed. We want to get rid of the multiple, extra triangles that disrupt the polygonal flow, since they add nothing to the mesh and drive the polygon count up.

To do this, the two points adjacent to the single, (circled) point that is part of the flow are welded to it. This removes four unneeded triangles, but creates another issue to be dealt with. Luckily, this is an easy fix too.

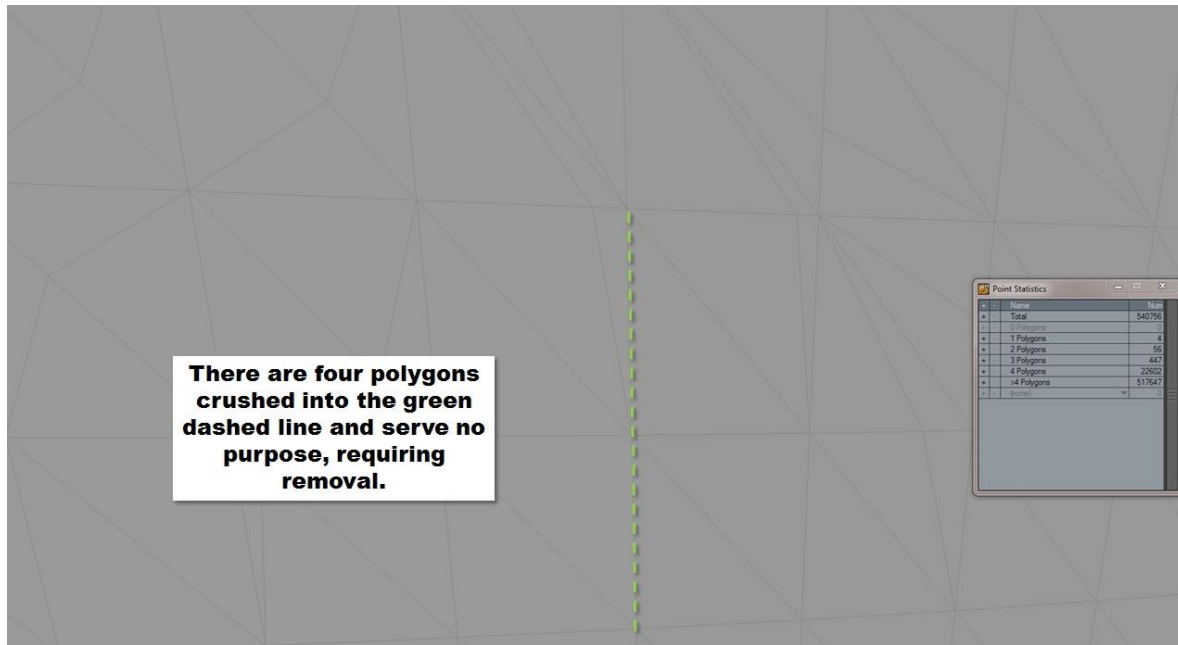


(Figure 4. Polys Destined for Removal.)

There are no free rides in 3D modeling, so it pays big dividends in time-savings to know what operations cause later issues.

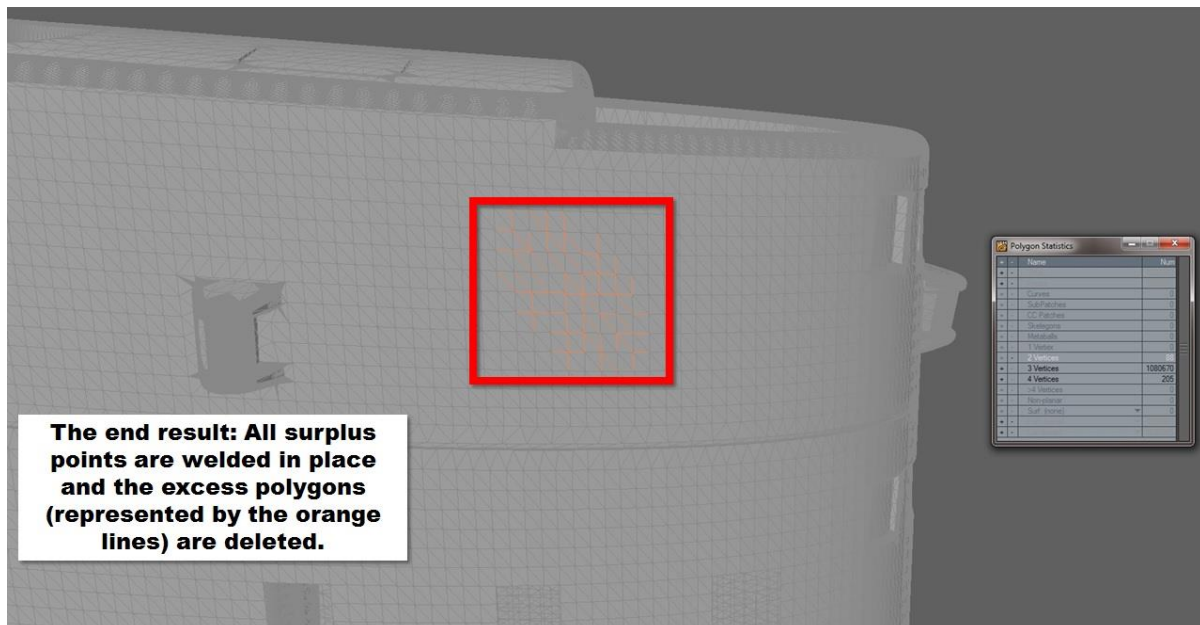
As shown in Figure 4, when the two points get merged into the center point, the four polys they define get “crushed” into the edge denoted by the green, dashed line running vertically along the Y axis. The mesh may look okay, but don’t let that fool you. Hidden inside are four, 2-point polygons requiring removal.

Note: 2-Point polygons are rarely needed when modeling, (they are often used as guides for strands of hair in a character model for example) and have to be identified and deleted. Left within a mesh, 2-point polys create rendering errors, prevent tools from working on surrounding polygons, and in the case of a 3D printed object, create non-manifold polygons which should be repaired within a slicer before printing.



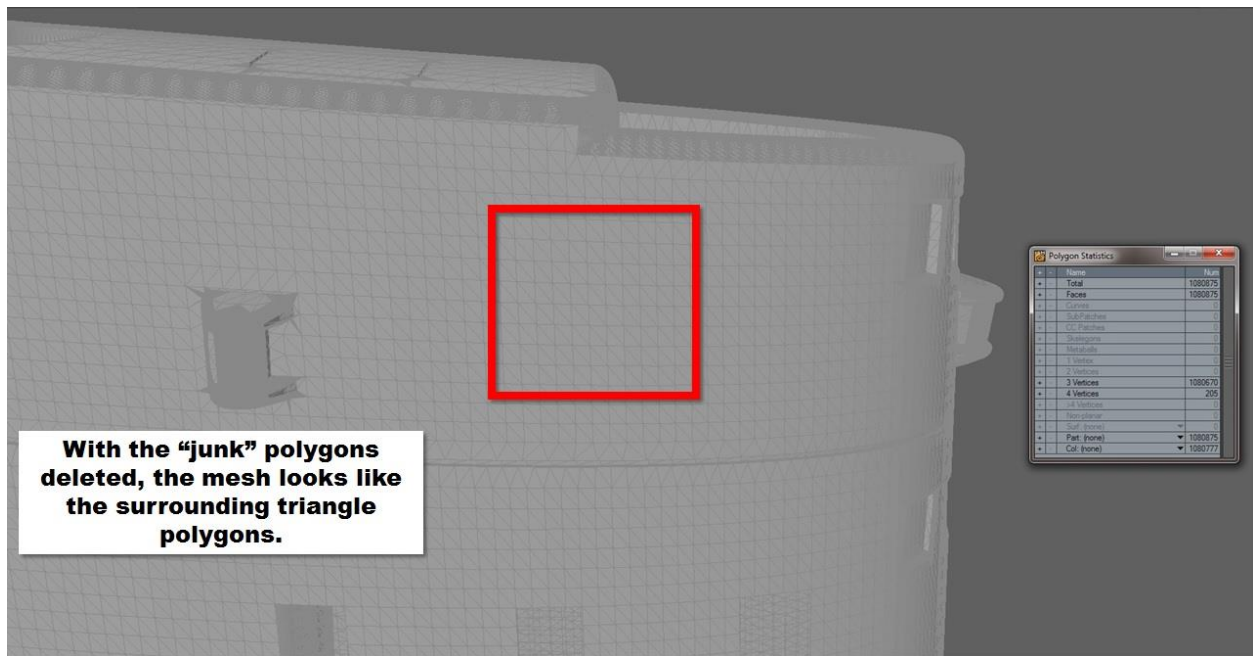
(Figure 5. Area of Concern)

Figure 5 shows the result after the three points are merged. (There's a triangle that needs to have its point welded at the top of the dashed line, but we're concentrating on the prior operation.)

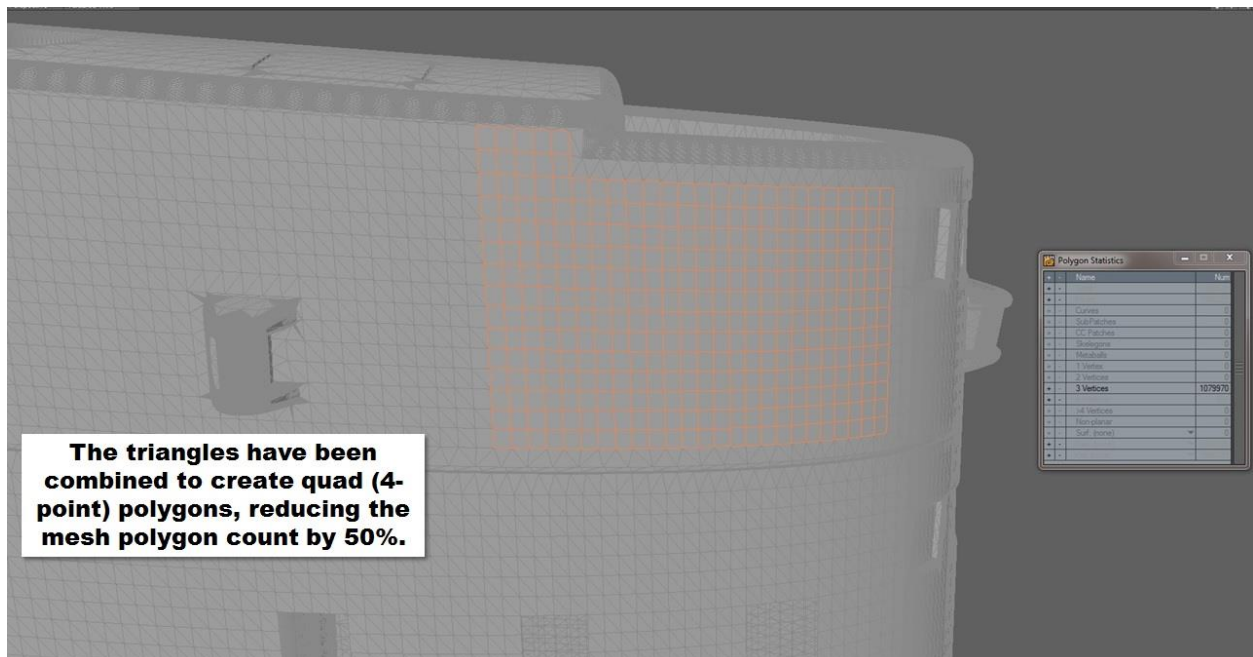


(Figure 6. All Excess Triangles Removed and Remaining 2-Point Polys Identified)

After welding the other unneeded points in the area, the Statistics panel is used to hunt down and identify 88 two-point polys hiding in the mesh. It is also used to select them. Once selected, pressing Delete removes them from the mesh.



(Figure 7. 2-Point Polygons Removed from the Mesh)



(Figure 8. Triangles Converted to Quad Polygons)

Figure 8 shows the result of using the Reduce Polygons command. With the selected triangles combined, the result delivers a smooth, all-quad mesh area with no loss of detail or faceting that would otherwise need to be sanded out of a 3D print.

To be precise, even if left as triangles, this mesh would look and print the same. The quad mesh makes changes easier, and takes up less computer memory improving performance. The polygon count of the sail could easily be reduced by 75%, but in this case, I could build it from scratch in a fraction of the time. Your mileage may vary.

Taking the polygon reduction process a step further, the columns of triangles without any details could easily be combined to create long, rectangles simplifying the mesh even more. When crafting a mesh, this is what I shoot for. Once the proportions or measurements are correct, details are added.

In the case of the Nautilus, which had many revisions over the years, (including a collision with the aircraft carrier Essex that took out the bridge all the way down to the front of the sail) it's a good plan to make a simplified base mesh of the sail and save it. If a later version of the boat is built, the file can be used to add new details without having to remove all of the older revision details and start over, saving a lot of time.

This is one (of many) mesh reduction techniques I've used over the years. In this case, the issues were blatantly obvious, and simple to repair, so I thought it might be worth a tutorial.

3D modeling is little more than combining simple techniques to create increasingly complex shapes. As the number of options at your disposal are added to your workflow, the faster you can achieve the desired results.