Curved Poly

version 1.3

Thank you for downloading Curved Poly!

This is an introductory documentation, please visit our resources online to get more:

- Full Documentation: **curvedpoly.com/guide/cpdocs**
- Video Docs (version 1.3): **Video Docs**
- Tutorials: **Tutorials Playlist**
- Youtube Channel: **Mushrooms Labs**

Don't hesitate to ask for more at **support.mushroomslabs.com**, or directly by email to **info@mushroomslabs.com**!

And please, don't forget to **rate us and write a review!**

Author and Developer: **Mushrooms Labs**

Project Site: **www.curvedpoly.com**

Author Site: **www.mushroomslabs.com**

Introduction

With Curved Poly Editor you can create models which use curves to shape complex curved polygons, which are then tessellated into meshes. Each Curved Poly Model can be tessellated into more meshes, with more or less real polygons, each one representing a different Level of Detail of the model.

Curved Poly Models

A Curved Poly is 3D Mesh made of Curves! It has curved edges and curved faces. Curved edges are used to connect vertices and curved faces are used to fill the content inside a set of Curved Edges.

Once a Curved Poly Shape is set up, it can be tessellated in many ways, originating different alternatives meshes from the same model with different Levels of Details.

Curved Poly Models can then be converted into meshes ... or they can be used in Play mode, at runtime, to access their full potential. Curved Poly models whose meshes are generated at runtime will need less space in your build, as each individual Curved Poly model actually uses far fewer vertices once saved in your scene or in your assets.

Curved Poly Tools

With our Curved Poly Editor you can access tools to generate or customize Curved Poly models on your needs.

The easiest way to model with Curved Poly is to start from a premade asset, where you can change its shape at your own needs. Once a model has been shaped, you can use the available tessellation Tools to fix your quality and performance needs, and there you go! Your new asset can be easily used to generate meshes assets, or it can be used the way it is, where the meshes assets are generated in play mode instead, during execution.

Once you have learned the basics, there are more tools you can have from Curved Poly Asset Store:

Curved Poly Shape Editor gives you access to a subset of tools and it's free. Note that the set of tools available with this version of the Editor allows you to reshape premade models, but you can't create new models from scratch. This make sense if you can access models made with curved poly. There is actually a plan to release on Unity Asset Store separated libraries of models made with

Curved Poly which you can take and reshape with this version of the editor.

Curved Poly Maker is the most complete set of modeling tools, giving you acces to everything in the editor. There are a few Tools Sets which are planned to be added to the editor in the future, and they all be add to Curved Poly Maker first.

Curved Poly Shape Editor PRO is planned to be released soon, and it's intended to be an intermediate level between Shape Editor and Maker. The actual idea is that you should be able to start with the free Curved Poly Shape Editor (having only 3 tools set), then you should be able to access more advanced features with Shape Editor Pro (a total of 6 toolsets) and then if you have Shape Editor Pro you will be able to access all Curved Poly Maker features at a lower price. Curved Poly Shape Editor Pro will give you all you need to start playing with parts (and materials), which can't be edited with the Shape Editor Only.

<< Back To Index

General Features

There is A LOT to say about Curved Polygons Nets. Here you can find a list of all the most interesting features you will find within **Curved Poly**

An extension to classical Meshes

Curved Polygons Nets are very closed to meshes. They share a lot of aspects with meshes. They have edges and faces, normals and uv coordinates. If you have experience in the field of modeling with polygonal meshes, you already know most of the things which you need to be known to work with **Curved Polygons Nets**.

Low Number of Vertices

Curved Polygons Nets use a little amount of vertices and a little amount of data compared to meshes. For example, the Sphere model is made with a total of 32 vertices (8 main vertices and 24 handles vertices). It is much less than every polygonal sphere you may have found in your experiences.

Scalability

Curved Polygons Nets must be transformed into polygonal meshes before being rendered, and this can be done at Run-Time. In order to do so, a Level of Detail must be provided. Such Level of Detail contains information about the number of segments and triangles to be used during the process. In this way, you can generate more or less detailed geometries from the same **Curved Polygons Nets** during execution, depending on your needs.

Mixed Models

Curved Polygons Nets can (since version 1.1 of the Shape Editor or version 1.0 of Curved Poly Maker) mix linear edges with curved ones. Polygons generated with lines are practically like regular meshes polygons. This allows the generation of optimal models which exploit Curved Poly features only in the part of your mesh that you need.

Segmentation

Within **Curved Polygons Nets**, each **Curved Polygon** and each **Curved Edge** may have applied a different set of geometric settings. For example, edges can be tessellated with a different amount of segments, and Polygons can be assigned different interpolation schemas (which are used to compute the position of vertices inside the polygon). This all work in an adaptive way, so you can take different choices through the model, which will be applied in a smooth way.

Elasticity

Many operators in Curved Poly works with special elasticity feedbacks which define the way the model reacts to transforms. You can choose among 4 different elasticity schemas.

UV Coordinates Groups

Having a little number of vertices make it much more easy to deal with uv-coordinates assignment. Furthermore, **Curved Poly** uses a structure called **Unwrap Group** which keep a set of uv-coordinates together. **Unwrap Groups** make it even easier any fix you wish to apply to uv-coordinates.

T-Junctions

Curved Polygons Nets support T-Junctions, like **Nurbs T-Spline**.

Mixed Polygons Sizes

Curved Polygons Nets support models where polygons have a mixed set of sizes. For example you may work using a mix of **Curved Triangles** and **Curved Quads**.

<< Back To Index

News (about version 1.3)

There are a lot of new features coming with version 1.3 of Curved Poly.

First of all. We have rearranged all the Tool Sets. Well, we used to call them Operators, but now they are called Tools Sets. The goal here was to simplify the access of each stage of modeling, which was becoming way too complex.

In this regard, the most important change is the introduction of Shape Tools, which replaces the old Pointings Operator and Selection Operator. Shape Tools has been designed to improve the way elements are selected and shaped, having the ability to interact with both single elements, and groups of elements. Shape Tools also integrate Hiding features and Backgrounds Tools, which previously had their own sections.

Another interesting change in Tools Set is repetition. This time, indeed, we have increased the number of tools which are available from every tools set. With few exception indeed, most of the tools have a selection and visibility section, and a parts list section, which work the same way everywhere.

Version 1.3, comes with a totally new Tools Set, which was previously in alpha stage, called Composition Tools. Composition Tools contains all the tools used to generated custom elements, previously grouped inside the Create Operator, which has been removed, but it also has tools which are designed to cut the parts of different models and merge them into new models.

Again, the Curves and Surfaces Tools Set alpha stage is closed, and now the Tools Set is fully operational.

Another important change is related to Asset Workflow, which is now accessible from the Editor rather than being accessible from Curved Poly Component Inspector. The most important difference is: curved poly models can now also be stored in scene and not only in the dedicated assets. Furthermore, we have finally removed the need of Edit folder, and Edit Assets, for Curved Poly Assets. The editor will still try to load data from Edit Assets when available, but will not create Edit Assets on newly generated contents any more.

Again, there has been some significant change in the way selections and interactive elements work. You can indeed select elements with one click, and there are three new types of elements which can be selected at once: parts, rings and loops.

On top of all this, the whole editor has a lot of minor changes in this version, almost all the Tools have been reworked and improved, they work faster, and the whole core of the project has been made more robust to errors. We have also improved the inline documentation which was introduced in version 1.2.5, and that is a good starting point to much up with all the minor changes.

<< Back To Index

Setup

In order to setup Curved Poly Editor, you need to download either the Shape Editor, or the Maker, from the Asset Store. Once you have acquired a package from the asset store, you can donwload it from the package manager. Then press import. After the Package has been imported, you will find a folder name **Curved Poly** within your Assets folder.

> More setup steps are available on the **full documentation page**, or you can watch the **setup video on youtube**

<< Back To Index

How to Start

> You can find more about how to start on the **full documentation page**, or you can watch the **how to start video on youtube**

There are multiple ways to start modeling with Curved Poly. Let's look for the Editor Window you can find in Tools, Curved

Poly.

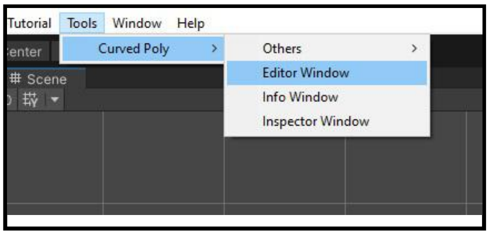
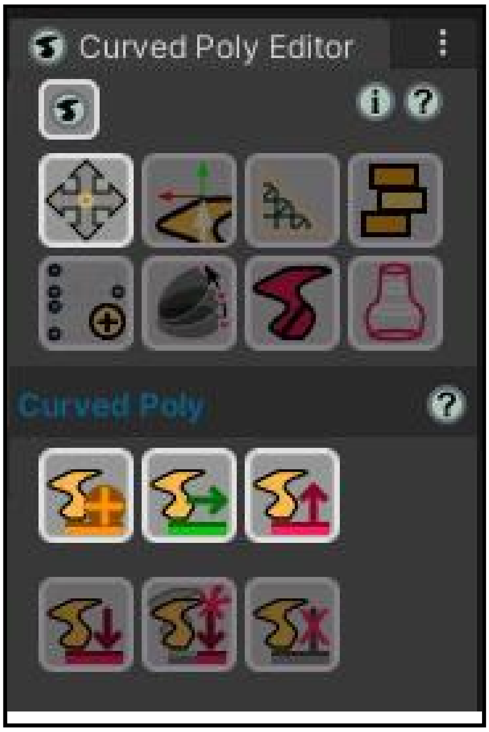


Fig. 6 Accessing the main Editor Window. From the same menu you can open any other window in Curved Poly.

Curved Poly Editor is organized in different Tool Sets, which are activated with the icons you can find here in the Editor.

When we have no Curved Poly models selected, the Game Objects Tools Set is the only one being active.

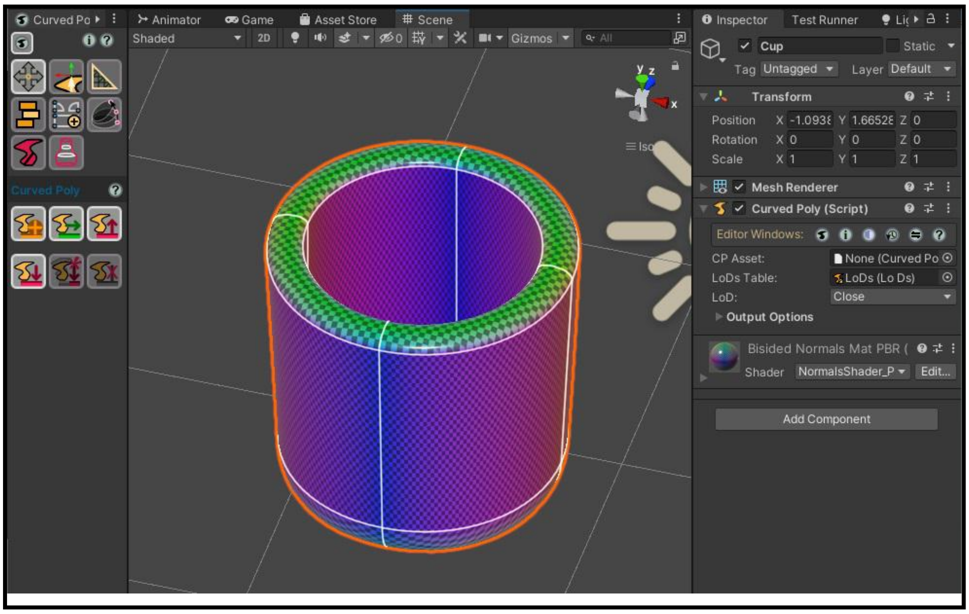


Curved Poly Editor Window, the first time you open it. When there are no Curved Poly Game Objects selected, most of the Tool Sets, the ones in the upper part of the editor, are disabled. Game Object Tools are available to start generating new Curved Poly Game Objects.

This ToolsSet has some available actions that are used to generate new Curved Poly Models. Let's start with **Import**. We can import a copy of any Curved Poly Asset available in the project into the scene. Curved Poly comes with a huge range of starting shapes and sample models you can get started with.

For instance let's copy the cup. A very simple shape.

Once a Curved Poly model has been selected, all the tools become available. **If you are using Curved Poly Shape Editor, only the first 3 tools are available instead.**



Cup

Tag Untagged Layer Default

Position X -1.0938 Y 1.66528 Z 0

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

CP Asset: None (Curved Po

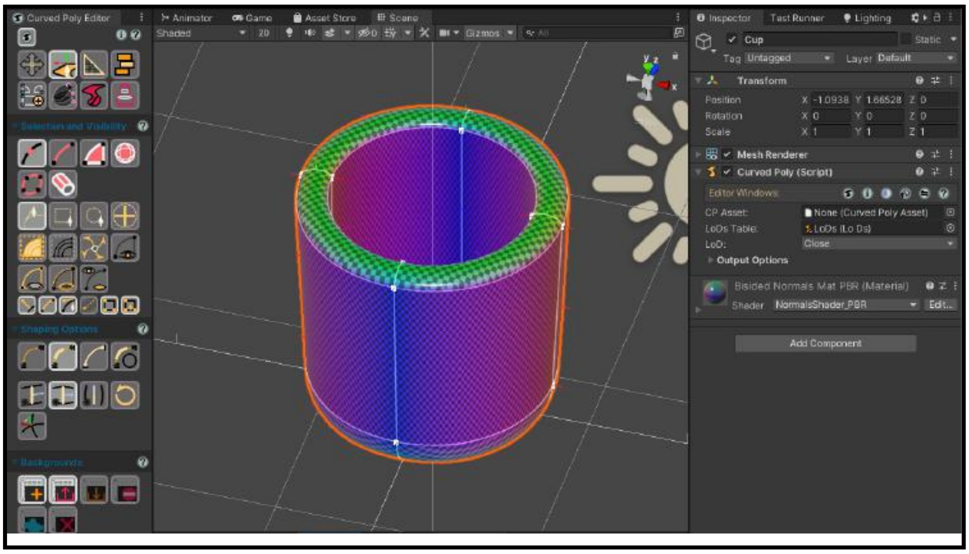
LoDs Table: LoDs (Lo Ds)

LoD: Close

Shader NormalsShader_P

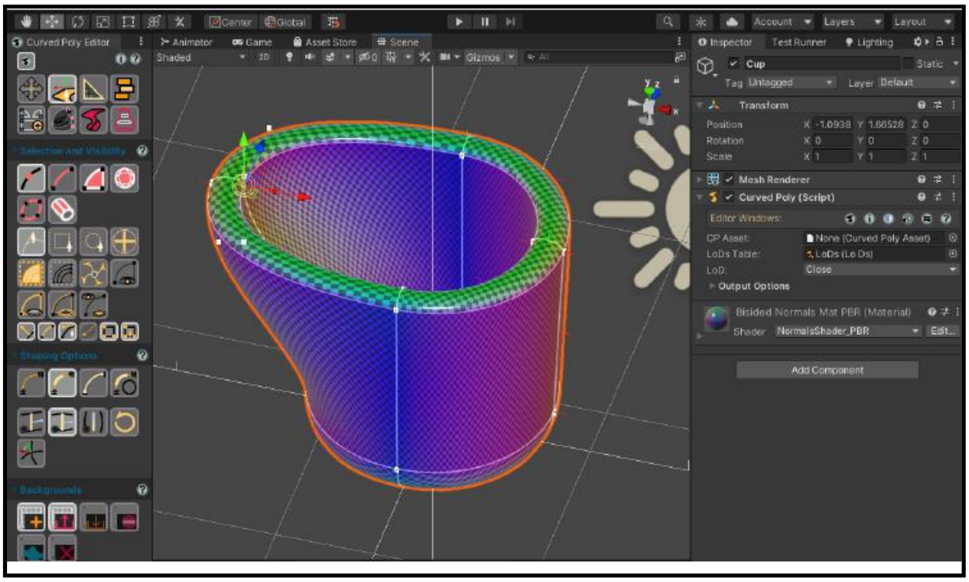
Once a model is loaded, the other Tools Sets become available, depening on your Curved Poly package you are using.

We can activate the Shape Tools. This is the tool to use when you want to edit the shape of a model. With this tool you can select different parts of it and change their position. For instance, make sure that the vertices option is active and click on one of the white cube standing for vertices in the scene. Here you have a Move Tool which can be used to change the position of a vertex. You can activate other vertices to modify other parts of the model.



Cup
Tag Untagged Layer Default
Position X -1.0938 Y 1.66528 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
CP Asset: None (Curved Poly Asset)
LoDs Table: LoDs (Lo Ds)
LoD: Close
Bisided Normals Mat PBR (Material)
Shader NormalsShader_PBR

If you press the the Shape Tools icon, the Editor will change and new tools will be offered to you.

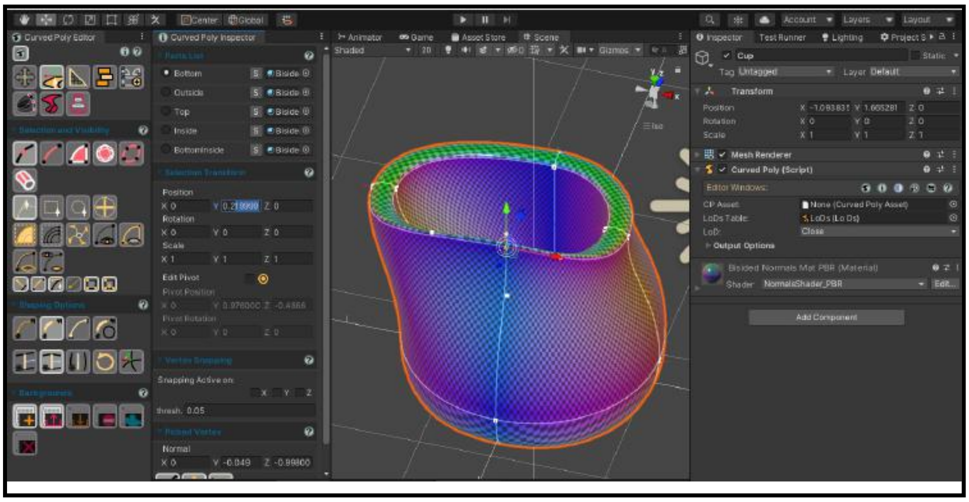


Cup
Tag Untagged Layer Default
Position X -1.0938 Y 1.66528 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
CP Asset: None (Curved Poly Asset)
LoDs Table: LoDs (Lo Ds)
LoD: Close
Bisided Normals Mat PBR (Material)
Shader NormalsShader_PBR

With Shape Tools you can start changing the position of vertices and the shape of curves, giving your starting model a completely different look.

You can see that the model is gaining a completely different shape from what we have started with.

While working with different tools, it is useful to access their full potential, through the dedicated inspector, which can be opened pressing the **(i)** button in the editor.



You can open the Dedicated Inspector from the Tools->Curved Poly Menu, or from the Curved Poly Component, or from the Editor Window. You will definitely need it at some point if you want to have full control other Curved Poly features.

For instance, we can manually change the position of a vertex, using the Selection Transform section options.

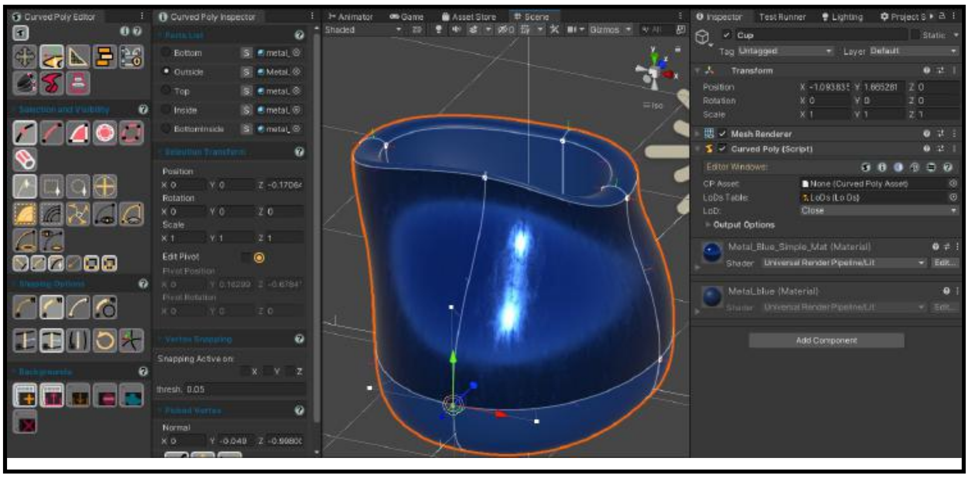


Fig. 14 With the parts list in the dedicated inspector we can easily interact with different parts of our model and,

for example, assign to them different materials or give them different names.

So, this was a fast glance at the editor. The rest of this guide will discuss every Tools set and every aspect of the Editor more in details.

<< Back To Index

Editor Interface

Curved Poly Editor is mainly organized into **Windows** which you can open from menu at **Tools>Curved Poly**

Once the main Editor Window is available, there are a total of 8 **Tools Sets** which can be activated and used for different purposes. Once you open the window, only **Game Objects Tools** become available, the other ones being useful only after a Curved Poly GameObject has been correctly created and setup.

<< Back To Index

Editor Windows

You can find more about editor windows on the **full documentation page**, or you can watch the **editor windows video on youtube**

In order to activate Curved Poly Editor we need to access the Editor Window. You can find it under Tools>Curved Poly>Editor window

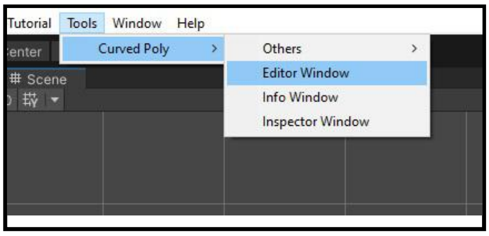


Fig. 15 Accessing the main Editor Window. From the same menu you can open all other windows used by Curved Poly Editor.

This window contains all the tools you need to work on your Curved Poly Models, and should always be opened while you edit them. It can be docked vertical on any side of the screen, or it can be docked horizontal, maybe in the bottom of the Scene Window, where all icons are placed in a more compact way without sections titles or support buttons.

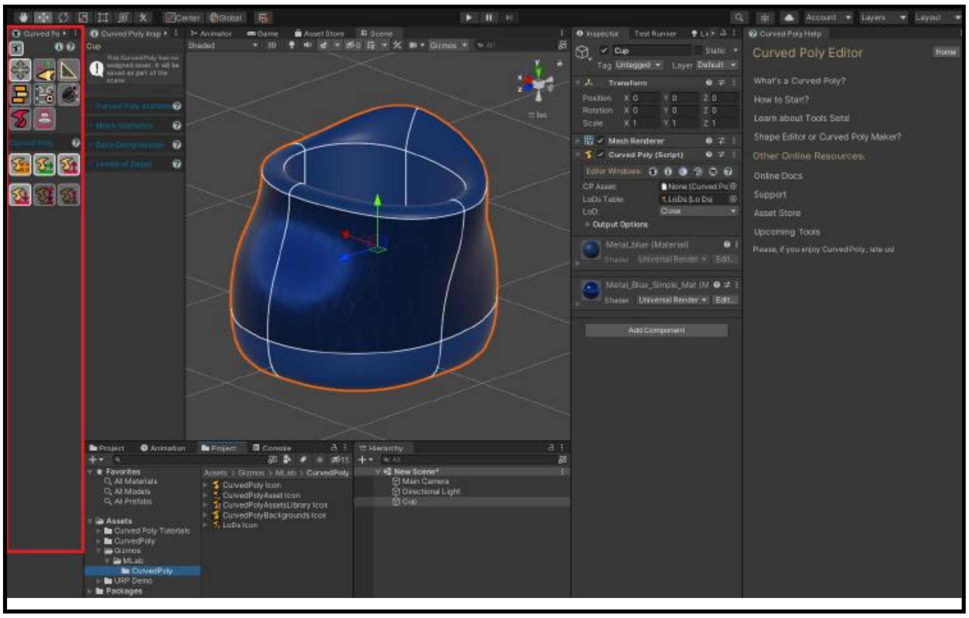


Fig. 16 Setup of Curved Poly Editor main Windows, with the Curved Poly Editor Window placed vertically on the left.

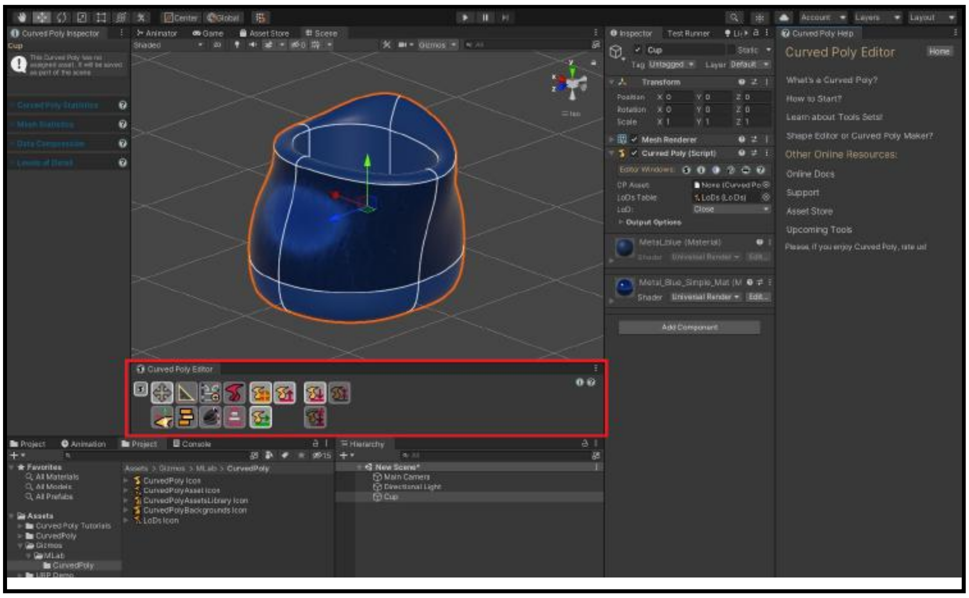
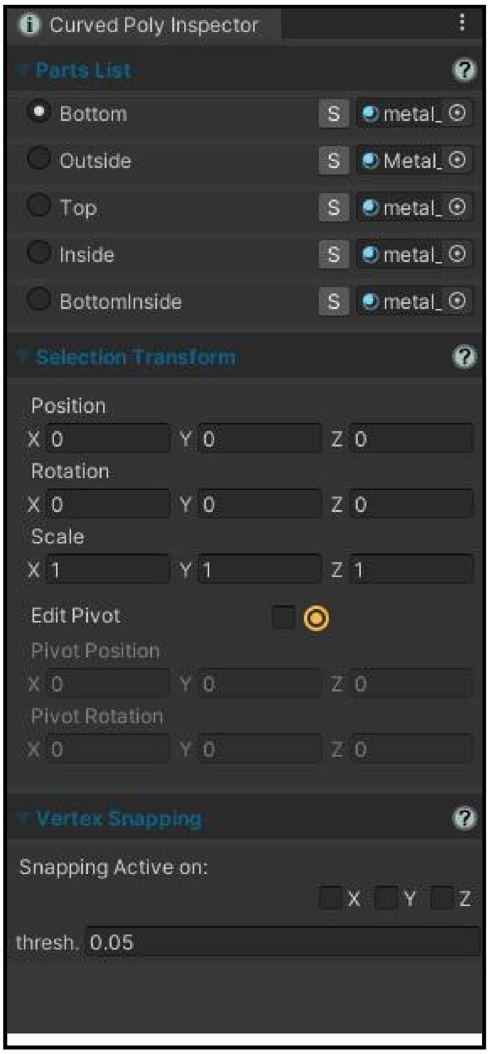


Fig. 17 Setup of Curved Poly Editor main Windows, with the Curved Poly Editor Window placed horizontally below the Scene.

Another important window we want to use is the Curved Poly dedicated inspector. You can access it directly from the Editor Window, or from the Menu. This window contains further instruments and details about the tools used in Curved Poly and having it available is somehow important to get full control other them.



Curved Poly Inspector

Parts List

- ● Bottom S metal_
- ○ Outside S Metal_
- ○ Top S metal_
- ○ Inside S metal_
- ○ BottomInside S metal_

Selection Transform

Position X 0 Y 0 Z 0

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

Edit Pivot □

Pivot Position X 0 Y 0 Z 0

Pivot Rotation X 0 Y 0 Z 0

Vertex Snapping

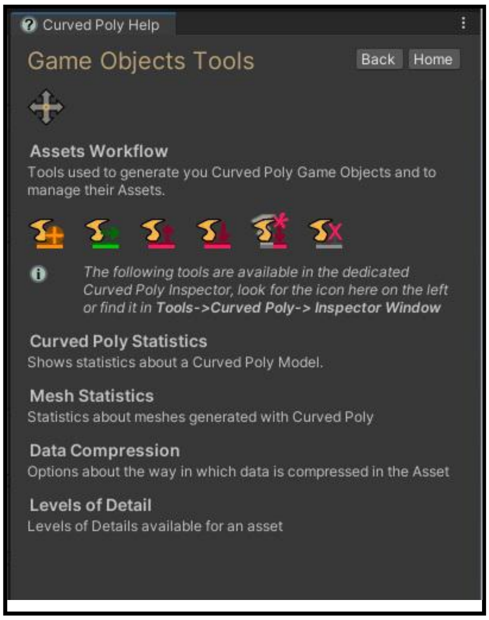
Snapping Active on: □ X □ Y □ Z

thresh. 0.05

Fig. 18 The Dedicated Inspector is required to access all the features of the Editor, even the ones which are not accessible from the main Editor Window. It's content changes according to the actual Tools Set selected in the Editor, and it can also change interactively depending on how you interact with the Editor in Scene. Here we are showing the default state of the Dedicated Inspector,

while the Shape Tools Set is opened.

Furthermore, if you are new to this editor, you should consider opening the in-line documentation, which can be accessed from the ?question mark icon. From this window you will be able to learn more about the tools, check the details of every parameter and also grab some tip about how specific tools shall be used.



Game Objects Tools

Assets Workflow
Tools used to generate you Curved Poly Game Objects and to manage their Assets.

The following tools are available in the dedicated Curved Poly Inspector, look for the icon here on the left or find it in ***Tools->Curved Poly-> Inspector Window***

Curved Poly Statistics
Shows statistics about a Curved Poly Model.

Mesh Statistics
Statistics about meshes generated with Curved Poly

Data Compression
Options about the way in which data is compressed in the Asset

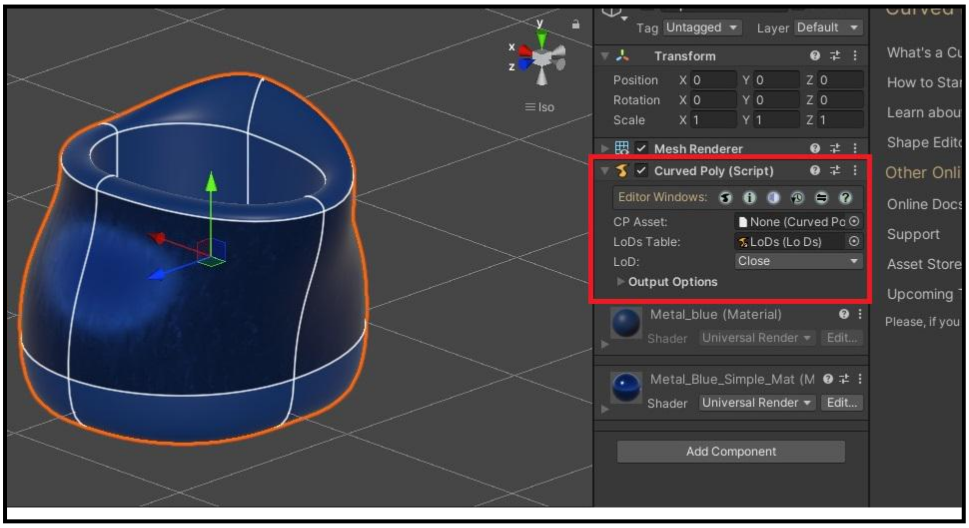
Levels of Detail
Levels of Details available for an asset

Fig. 19 This is the main Curved Poly Help window, an interactive window showing a description of tools used by Curved Poly. Its contents change according to the tools you are using, and you can force a specific content both from the Editor Window, or from the Dedicated Inspector, by hitting the question mark icon.

Once your interface is up, the best thing to do is to import a premade model into the scene using the import button. While you work on the model, the documentation window will keep update according to the tools you are using, showing you how to use them. And you can press on the question mark button to know more about the tools around.

On top of the Editor Window you will find the main tool sets, each one being a set of tools which are useful in a different stage of modeling. Clicking on one of this icon will change the content both of the editor window and of the dedicated inspector. In some cases you may give your editor window a different position: since version 1.3 of curved poly we have reworked the structure of this toolset; the goal was to limit the number of times you need change toolset in order to operate on your shapes.

There are again a few more window you should consider, but they are not mandatory. You can access a full list of windows also from the main Unity Inspector under an instance of Curved Poly Component.



Tag: Untagged
Layer: Default

Transform
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

☑ Mesh Renderer

☑ Curved Poly (Script)
Editor Windows:
CP Asset: None (Curved Po
LoDs Table: LoDs (Lo Ds)
LoD: Close
Output Options

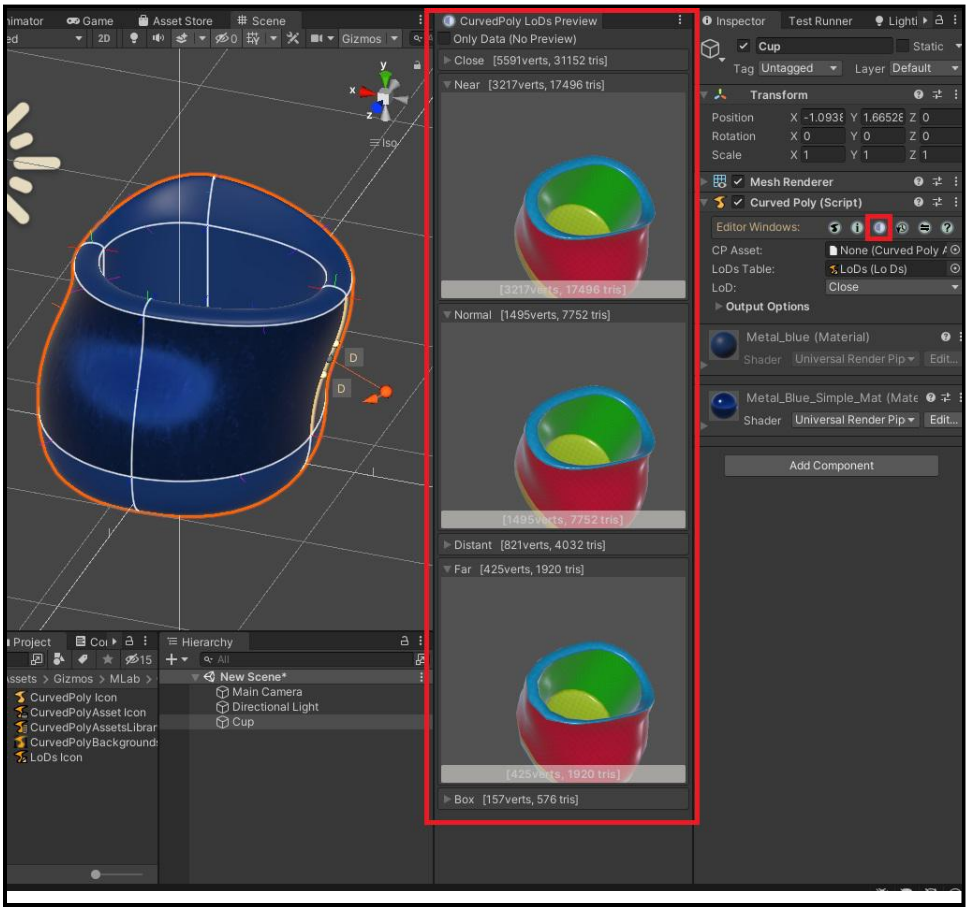
Metal_blue (Material)
Shader: Universal Render

Metal_Blue_Simple_Mat (M
Shader: Universal Render

Fig. 21 The Curved Poly Component, in the main Unity Inspector. At the beginning of the component, you can find a list of icons, one for each available window. This is an alternative way to access the editor interface.

Here we can open the history, with the full list of changes you made on the model which can be undone and the settings window where you can find a few settings.

Last but not least, there is the LoDs preview window which is used to give you a preview of your model as it is tessellated with different Levels of Details. The available LoDs shown in the window are the ones used by a Curved Poly component, which can be customized as we will see later in a dedicated section. The previewer is useful, but not strictly necessary since you can preview your model on different Levels of Details from the inspector choosing a different LoD for the model itself. It may be also useful to active the Selection Wire option on the Scene to see how exactly the model is tessellated on different LoDs.



CurvedPoly LoDs Preview

☐ Only Data (No Preview)

Close [5591verts, 31152 tris]

Near [3217verts, 17496 tris]

Normal [1495verts, 7752 tris]

Distant [821verts, 4032 tris]

Far [425verts, 1920 tris]

Box [157verts, 576 tris]

☑ Cup ☐ Static

Tag Untagged Layer Default

Transform

Position X -1.0938 Y 1.66528 Z 0

Rotation X 0 Y 0 Z 0

Scale X 1 Y 1 Z 1

☑ Mesh Renderer

☑ Curved Poly (Script)

Editor Windows:

CP Asset: None (Curved Poly A

LoDs Table: LoDs (Lo Ds)

LoD: Close

Output Options

Metal_blue (Material)

Shader Universal Render Pip Edit...

Metal_Blue_Simple_Mat (Mate

Shader Universal Render Pip Edit...

Assets > Gizmos > MLab >

CurvedPoly Icon

CurvedPolyAsset Icon

CurvedPolyAssetsLibrar

CurvedPolyBackground

LoDs Icon

New Scene*

Main Camera

Directional Light

Cup

Fig. 24 Curved Poly LoDs Preview

<< Back To Index

Tools Sets

Let's have an overview of Curved Poly Tools Sets. You can find them on top of the editor window. A Tools Set, as the words say, is a set of tools. Clicking on one of the icons, changes the overall set of available instruments, both in the rest of Editor Window and in the dedicated inspector. In versione 1.3 we have reworked the structure of the Tools Sets, which were previously called Operators. This has been done by regrouping tools in a different way. The idea here was to make the overall modeling process more easy, and to minimize the number of times a switch between tools sets is necessary. So, each tools set stays now for a precise moment of a modeling workflow. When we start, only the Game Objects Tools are available. But after creating a new Curved Poly Game Object, more Tools Sets become available, depending on the Editor Plugins you have installed. With Curved Poly – Shape Editor, the tools available are shape tools and tessellation tools. With Curved Poly – Maker all the other tools become available. We are planning to release separated packages with separated groups of tools sets in the future, but for now the only way to have them all is by getting Curved Poly Maker.

With Shape Tools, we can take control over the shape of models, changing the position of vertices, and modeling the edges, where polygons will shape according to the shape of their edges. Shape Tools uses automated algorithms to control how the model react to shaping actions, which can be changed during the process of shaping. It also allows to work on backgrounds, which can be placed behind the model, and support you during the modeling process.

Tessellation Tools are used to work on tessellation processes. Where you can control how much each edge should be tessellated, and the polygons will smoothly tessellate according to their edges. The process requires the assignment of tessellation hints, which control the way edges and polygons are tessellated on different levels of details, which can even be changed at runtime.

Composition Tools is one of the most recent Tools Set in curved Poly. Here you can combine different parts, coming from premade models you have, or generated with the primitives creation tools. You can also break parts which are connected together, or remove parts you don't need.

Materials Tools allow to work on the way materials are applied to your model. Here you can split a model into more parts, where each part can be assigned a different material. And you can also work on unwraps or uv coordinates.

Advanced Shape Tools is an advanced version of Shaping Tools, featuring the same tools and more interesting instruments to take control over the shape of models.

Edit Tools contains all you need to customize your shapes, by cutting elements you don't need, create new elements from scratch or split elements into more little elements. Here you can also control alignments, which are used in many models to handles aligned, the sharpness of vertices, which is used to create models with sharp edges, and the type of edges which can be lines, or curved edges.

Last but not least, we have Curves and Surfaces tools. This is recent tools set, designed to help in the generation of custom primitives. Here you can setup curves which are used to shape surfaces. Once your surface is ready, you can press on Convert into Curved Poly to start modeling your surfaces as a regular curved poly model.

<< Back To Index

Assets Workflow

You can add a Curved Poly Model to your Scene by adding a Curved Poly Component to any Game Object in Unity.

Curved Poly Models can then be generated and stored in a Unity Project in many ways. Section **Assets Workflow** has an in depth covering of all the possibilities.

On the **Curved Poly Component** there are different ways to work on the result of Curved Poly Processes. You could just use Curved Poly has a mesh generator, or you can use Curved Poly at runtime. Whatever is your path, the Component offers many options and informations about how meshes are generated from the model.

<< Back To Index

Assets Workflow

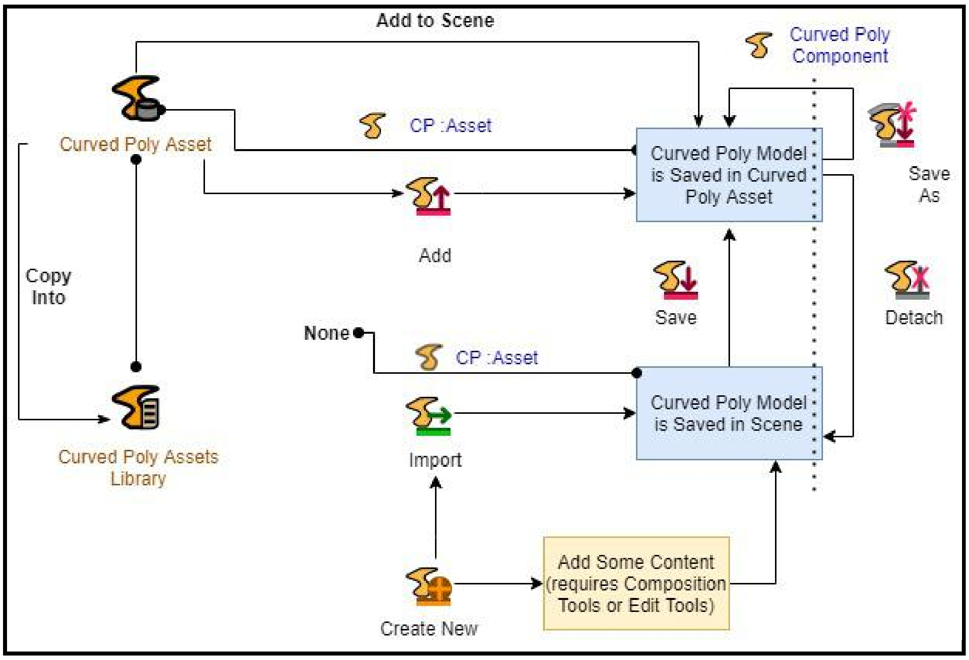

Add to Scene
Curved Poly Component
Curved Poly Asset
CP :Asset
Curved Poly Model is Saved in Curved Poly Asset
Save As
Add
Copy Into
Save
Detach
None
CP :Asset
Curved Poly Model is Saved in Scene
Curved Poly Assets Library
Import
Add Some Content (requires Composition Tools or Edit Tools)
Create New

Fig. 32 Full Assets Workflow Diagram

You can find more about assets worflow on the **full documentation page**, or you can watch the **assets worflow video on youtube**

Note: If you have asset made with previous versions of Curved Poly Editor, you may be interested in reading about **what's happening when assets are loaded with version 1.3**

There are more ways in which Curved Poly Editor stores data into your project. The Assets workflow has been completely reworked, and significantly improved, in version 1.3, so, this section of our documentation is to be considered new also for people who have already experienced any previous version of Curved Poly. Let's start with the basics. Whereever you want to start modeling something, you could start with a premade shape. There are two main ways to do so: from Game Objects Tools, you can press **Import** and grab an Asset. *In this case the asset is imported as a copy of the original model*. A game object is generated in scene, with a Curved Poly Component. The Curved Poly Component has assigned an Asset whose name is none. This means that the model you cloned is stored within the Curved Poly component, which is part of the Scene, and it doesn't refer to any specific asset.

On the other side, you can use **Load** in place of Import. Load does pretty much the same thing, with the huge difference that once a model is loaded, it's Curved Poly Component is directly referencing the original asset itself. This means that the model will not be saved in the scene. And that any change on the model will affect also the original asset, and every other object using it.

So we have **models stored in scene**, and **models stored in assets**. It is possible to create new Assets from both of them. Creating new assets is good since you can reuse them on more objects. Let's say we have made some changes to our model and we want to store the changes in a new asset.

When we press **Save**, a new asset is generated with a preassigned progressive name, but you can move it where you want, and rename it as you wish.

When you select the asset in the project, the Inspector shows a preview of the model itself. Here we have a third way to add models to the scene, by clicking on **Add to Scene**. Add to Scene generates a new model with the same name of the Asset added, and the new object is referencing.

Once the object using the asset is in scene, you may need to break the link with the original asset. There are two ways to do so. We can press on **Save Asset As New** to generate a new asset different from the first one. The new asset is now independet from the first one, and can be reshaped apart.

Another possibility is to detach the asset from the object, with **Detach From Asset**. In this way the asset is cloned on the Curved Poly Component, and the reference on the Curved Poly Component becomes none, making the models to be stored in scene, rather as part of a Curved Poly Asset.

There's another important thing about Curved Poly Assets. They can be collected into libraries, making it easier to share them among more projects. In order to do so, we go on Project and press on Create, Curved Poly, Curved Poly Assets Library. This generates an empty library. Now you can select one or more Assets, and there is a button in the inspector saying Copy into library. We click on **Copy into library**, and we can choose which is the library we want to send our objects to. The assets are copied into the library, so in this case we still need to decide if we need to keep the original assets, or if we wish to dispose of them and keep only the copy we made in the library.

<< Back To Index

Game Objects

> You can find more about curved poly game objects on the **full documentation page**, or you can watch the **curved poly component on youtube**

To add a curved poly model in scene, you need to have a Curved Poly component active on one of your game objects. As we have seen, this can be done with Game Objects Tools from Curved Poly Editor. Or, it can be done by creating a new Curved Poly with Create Other, Curved Poly, New Curved Poly. Adding a Curved Poly component adds a *Mesh Renderer*, used to render the mesh generated by curved poly. There is also an *hidden mesh Filter added*; the Mesh Filter is hidden since you cannot assign a mesh to it. Curved Poly will use it to assign the proper mesh to your model. The Mesh Filter is still accessible by code in case you need it.

The Curved Poly Component has three important parameters: the first one is the Asset used (**CP Asset:**), which may be unassigned, or None, meaning that the model is stored in scene, rather than being stored in a shared asset. For instance we can drag and drop here a Curved Poly Asset. And we can remove the Asset again to remove the reference. Again, in this case we have created a perfect copy of the original asset, which will be saved withing the scene.

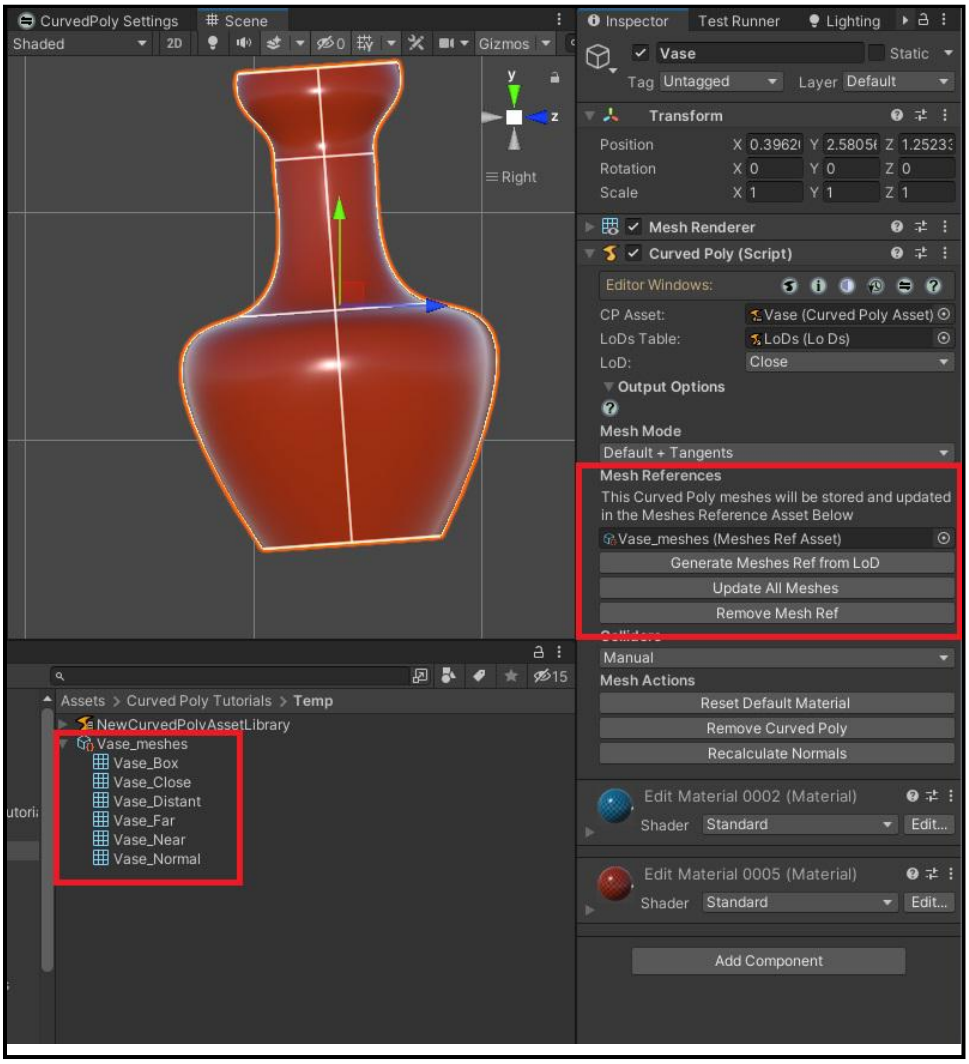
The second parameter is the Levels of Details table(**LoDs Table:**), that is the table which describes the Leveld of Details which can be used on the model. LoDs table can be customized, and we are going to show this in a later section, but for the rest of this presentation we are going to keep using the default LoDs asset.

And the third parameter here is the exact LoD choosen (**LoD:**), which must be one of the LoDs available in the table. As you can easily check on your own, changing the LoDs will upgrade the model with a different amount of polygons.

The component has also some advanced tools in the Output Options sections. In order to understand this section, keep in mind that Curved Poly Models are not exactly visible in scene. What you can see here is a regular Mesh which is generated by Curved Poly. These output options define how the Mesh is generated.

The first control is the Mesh Mode. This defines which mesh data are going to be generated from the model. By default it should be set to Default plus tangents. In version 1.0 of Curved Poly tangents were not generated on the model, so if you select Default you are going to miss tangents. With no UV the mesh will be missing both uvs and tangents. In version 1.1 we had added an option to generate automaticcally uv2, but recently Unity has added some similar automation which integrates well with Curved Poly, and you don't need this any more. Use this if you are using Curved Poly on a Unity version previous to 2019 I say, and you need uv2.

Talking about integration with other plugins and tools, you should consider to use Mesh References in some situations. Indeed, by default Curved Poly generates Meshes at runtime. Since the meshes are generated at runtime, there is no way to feed them to other plugins or components you use in the editor, but you can do that using scripts. In case you can't do your own scripts, Mesh References are your best option. When we press on Generate Meshes Ref from LoD, we are asked to place a new asset somewhere in the project. The asset will contain a set of Regular Unity Meshes, one for each LoDs, which can be used everywhere in the project. While the link between the Meshes Ref Asset and the CurvedPoly component is active, any update to the model will be applied to the mesh matching the assigned LoD, and Curved Poly will keep using the meshes in the asset even in play mode, upgrading them when necessary. There is also the option to Update all the Meshes at once. It is also possibile to remove the Mesh Ref from the Curved Poly Component. In this case the Meshes Ref Assets will keep being in the project, but the meshes will not be updated any more.



Vase Static
Tag Untagged Layer Default

Transform
Position X 0.3962(Y 2.5805(Z 1.2523:
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1

Mesh Renderer

Curved Poly (Script)
Editor Windows:
CP Asset: Vase (Curved Poly Asset)
LoDs Table: LoDs (Lo Ds)
LoD: Close
Output Options
Mesh Mode
Default + Tangents
Mesh References
This Curved Poly meshes will be stored and updated in the Meshes Reference Asset Below
Vase_meshes (Meshes Ref Asset)
Colliders
Manual
Mesh Actions

Edit Material 0002 (Material)
Shader Standard

Edit Material 0005 (Material)
Shader Standard

Assets > Curved Poly Tutorials > Temp
- NewCurvedPolyAssetLibrary
- Vase_meshes
  - Vase_Box
  - Vase_Close
  - Vase_Distant
  - Vase_Far
  - Vase_Near
  - Vase_Normal

Fig. 40 A Meshes Reference Asset applied to Curved Poly Component.

Then, below here, we have the Colliders options. This is interesting only in case you are using a Collider on the same GameObject, and you want to make changes to the CurvedPoly Asset at runtime. In this case, Curved Poly can indeed update the parameters of the Collider at run time. For Mesh Colliders, the collider mesh will be assigned by choosing one of the available LoDs on the same model. So, playing with LoDs we can generate a very details model with a more simplified shape in its collider. Both the model and the and the Collider mesh can be updated if you change the asset on the model at runtime.

Again, below colliders we have a few mesh actions. We have already seen that we can rest a model to Curved Poly Default Material here in once click. We can also remove the Curved

Poly Component. In this case a visible Mesh Filter will replace the model, with a Mesh assigned which depends on the last LoD which were selected on the model before removing Curved Poly Component.

Before removing a Curved Poly, or before detaching a Meshes Ref, it may be useful to recompute normals. What it happens here, is that Curved Poly uses Unity recompute normals algorithm to change the normals. This is usually a downgrade, since Curved Poly normals are generated on Curved Poly data, rather than Mesh Data, and therefore are more precise. That said, we may have specific situations in which Unity Normals are better than Curved Poly normals. Keep in mind that any change to the model will use again Curved Poly algorithms to compute normals, so this is an action which make sense only if you are going to Remove Curved Poly Component or Remove a Mesh Ref asset from it, because in that cases you will get a mesh which is not affected by Curved Poly any more.

Let's say we have already open the Editor and the Dedicated Inspector, and shown in the previous section. From Game Objects Tools, we can overview a few more tools related to a Curved Poly Components. This sections were previously part of of the main Curved Poly Inspector, but we moved them here to make it a bit more simple.

Here you can find a few statistics about the model, and about the mesh generated from the model. There is also the option to overview the Levels of Detail used on the model, which can be changed here: but keep in mind that any change here will affect any model using the same LoDs asset. We also suggest to avoid making changes on the original default LoDs asset, where you can create a new custom LoDs, or clone the default one, if you need to make changes.

Last but not least, we have a Data Compression Section. Here you can work on parameters which define how data is compressed in Curved Poly models. Indeed, vertices, normals and uvs can be saved with different resolution. Increasing the resolution, increase the precision but also increases the size of the asset. Take for instance the Vertex Precision: this allows you to define the resolution used to save vertices coordinates, where the default MILLI stays for one milli part of the Unit, which for most real curved poly models is pretty much enough. The default settings are usually fair enough, but in case you fill you need more precision, you can refer to the inline docs which explain better the meaning of each parameter.

<< Back To Index

Assets Upgrade to 1.3

In version 1.3 of Curved Poly we have made some significant change in the way assets are managed. If you have been using Curved Poly before version 1.3, you may need to upgrade your assets to version 1.3.

> DON'T WARRY! Your models made with previous versions of Curved Poly are going to work also in version 1.3, and usually there is no real need to make an upgrade, especially if you don't plan to edit them again. If you try to edit them, the editor will automatically upgrade them. **But** it is strongly recommended to make a backup if you want to edit them again: with a backup you will be able to recover your models and make a manual upgrade, in case something goes wrong, using the procedures which are shown in this page. That said, there are good reason to upgrade to version 1.3, so you could consider running the manual procedures in any case.

So, what are the main differences with the old assets?

- **We don't use the Edit folders and Edit Assets anymore**. This makes the assets workflow much more simple and intuitive. If you have edit assets made previous to version 1.3, Curved Poly is still capable to read them the first time the asset is accessed from the editor. After the first access, the asset will upgrade to version 1.3, and the Edit Assets will become useless. Please, don't remove from the project the old edit assets before opening the assets from the editor, because that would result in a loss of data.

- **We can now save a Curved Poly model in scene**, without having it stored as a shared asset. As a consequence of this **you can now store and edit a Curved Poly models also in prefabs**. We have tested the prefabs editor with Curved Poly Editor, and we strongly believe it's a very great way to work on Curved Poly Models.
- Curved Poly Assets and Models now contain also edit data. Such edit data is cut from the assets when the project is built for platforms different from the Unity Editor.

How to upgrade all the assets and once?

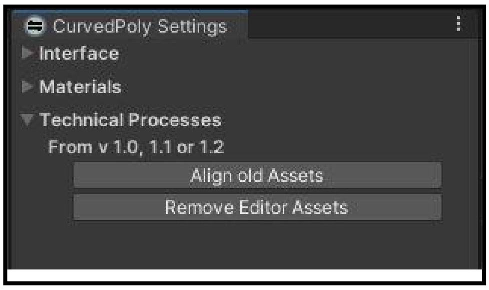


Fig. 43 Assets Upgrade tools in Curved Poly Settings

Yes, you can upgrade all the assets at once. This will make all your assets to benefit from the new Asset worflow. Let's see how to do this. The main advantage of this is the removal of all the old Edit Assets the project.

- **Make a Backup** please: if you keep a copy of all Curved Poly Assets, it will be possible to recover the data if something goes wrong. Otherwise your models may be broken and it will be not possible to get it back. **Remember:** the best way to make a backup in Unity is through the Export Package option on your assets folder. When you do this, double check that all the edit assets and folders are selected together with your assets.
- **Look of Curved Poly Settings window** : you can do this from Curved Poly Component, within the Unity Inspector, of from the menu at **Tools>CurvedPoly>Others>Settings Window**
- **Look for Technical Processes** : there are two actions here, **Align old Assets** and **Remove Editor Assets**. They both upgrade all the Curved Poly Assets in the project to version 1.3. The difference is that the second one also removes all the Editor Assets, while the first one keeps them.

Can automatic conversion break a model? What to do in that case?

We have tested more and more times this procedures on all the premade models and primitives which you may have found. If your models are shaped versions of that ones, made with Curved Poly Shape Editor, it's pretty much safe that everything should go well. If your models were made with Curved Poly Maker, let's repeat again that we have made a lot of tests of this, on model combining different aspects of Curved Poly, in many different ways. We, of course, cannot make test for all the custom shapes you may have done, in case something goes bad we are willing to help you fix the thing. Contact us at support pages or write to info@mushroomslabs.com.

<< Back To Index

Common Tools

You can find more about common tools on the **full documentation page**, or you can watch the **common tools on youtube**

During almost every moment of a modeling process, you will need to have something to **select which elements you want to interact** with in your model, and maybe you may need to **change often what to show on the models**, in order to succesfully work on every element on it. For this reason, there are common tools which are available from almost every **Tools Set**.

Together with the section of the **Selection and Visibility Tools**, on the same Tools Sets you will find the **Parts List** (that appears in the dedicated inspector).

<< Back To Index

Selection

This chapter covers common tools which can be found in almost all the Tools Sets. These tools can interact with almost everything in the editor, and are therefore very important to be learnt.

In order to interact with a model, we need a way to pick its elements. For this reason there are 6 different selection modes you can choose among.

With **Vertices Selection Mode** , you can interact with the vertices of your model. For example you can click on a single vertex to start reshaping it.

Edges Mode is used to select single edges, for instance clicking everywhere on them. Again, once an edge is picked, there are more options to interact with it.

Polygons Mode is used to pick curved Polygons, again by clicking on the polygon surface.

The other options apply to specific groups of elements.

A **Loop** is a loop of chained edges. In order to pick a loop, we need to click on one of its edges. The condition for two edges being in chain is that they meet at the same vertex, and that they are almost aligned on that vertex. For example, loop is useful to shape at once loops of edges on cylinders, or on other similar primitives. Depending of the edge you start with, the loop may be closed or open.

Opposite to the loop is the ring. A **Ring** is a chain of polygons. In order to pick a ring, we need to select an edge, and the polygons attached to that edge will be selected first, then the chain is propagated to more polygons if possible. The condition here for two polygons being chained is to have a common edge. The chain is propagated only on curved quads, through opposite edges, and stops if any polygon is found not being a quad, like a curved triangle. Depending of the edge you start with, the ring may be closed or open as well.

In **Parts Mode**, we can click on a polygon to select all the polygons being in the same part. This makes sense only for models which have already been splitted into parts. If you want to try this, you should start from some of the premade models like the moka or the cup. An interesting way to look for models with more than one parts, is looking in their asset inspector. Here models with more parts are rendered using different materials.

In the previous examples, we have seen how to pick one element at a time, but also multiple selections are supported.

Below the selation modes, you can find the group of selection Tools.

By default, the **Pick Tool** is selected. With this tool we can pick a single element, by clicking on it, and, as we have seen before, one or more elements maybe selected, depending on the Selection Mode in use.

With the **Rectangle Tool** we can select more elements at once. by dragging a selection rect on screen. Again, the overall selection depends upon the selection Mode chosen. For instance, if Vertices selection Mode is active, and the rectangle is dragged over the two vertices at the extremes of an edge, then both the vertices, and the edge, will be selected. Some thing for the vertices of a polygon.

The **Circle Tool** works in a similar way. We can drag the circle over the elements we want to be selected. And we need to press the **P** or **O** keys on the keyboard to reduce or increase the size of the circle.

For all the selection tools we have the option to active the cumulative variant. When **Cumulative** mode is disabled, any picking or selcting action will clean the selection. Whene Cumulative variant is enabled, elements can be selected in more steps, since the selection is never cleaned. So, for instance, in Vertices Mode, with the Pick Tool in Cumulative variant, we can select, one by one, all the vertices of a polygon to get it selected.

There are also a few actions to know about selection. You can select everything or remove everything from the selection, with **Select All** or **Unselect All**. It is very practical to know that you can execute the same actions with the **S key** for Select All, and with **Shift plus the S** key for the UnSelect All.

Another usefull action is **Grow**. Grow is used to add to selection every elements adjacent to elements which has already been selected. For example, if we select a vertex and press Grow, every edge or polygon attached to that vertex will be selected; and we can click grow more times, to propagate the selection on more and more elements.

<< Back To Index

Hiding and Visibility

Together with the Buttons for instance selection, we can find the controls required to hide or show elements. Indeed, elements can be hidden, in order to temporarily remove them from the editor. Then hidden elements can be shown back to be visible again.

The most intuitive and simple way to hide elements is to select them first, and then click on Hide Selected. We can do this more and more times if we need it. And when we want to restore elements which were hidden, we need to click on Show All.

There is also the opposite workflow. We can select a group of elements we still want to see, and hide everything else. This can be done by pressing on Hide UnSelected. Again, when we want to restore elements which were hidden, we need to click on Show All.

There is also an option to hide what can't be seen. Where, what can't be seen depends on the actual orientation of the camera. So, for instance, if we keep this orientation in the Scene Window, and we click on Hide Invisible Elements, it looks like nothing is happened; but if we change the orientation of the camera, by rotating around the model, we can see that all the elements behind what we were watching has been hidden.

Below the Selection and Hiding Actions, we can find a set of visibility options, which controls the visibility of the editing gizmos, and the operativity of some of the other tools.

When **normals** is selected, you can see normals on your model, as little lines, and you can also edit them. On a Curved Poly Models, normals can be assigned to vertices and edges. Edges normals appears as lines exiting from the center of the edge, and they only exist on curved edges, so they are not available on lines.

Edges allows to see all kind of edges on the model. Edges are still visible, reguardless of the edges settings, if edge mode is enabled.

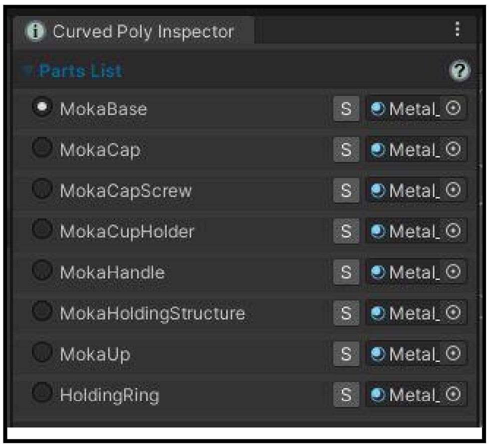
Handles shows the handles which control the shape of curved edges. Handles are available both from the vertices they are attached to, and from the edges they belong to, but they are not accessible on all the Tools Set.

With **Show Hidden Controls** we can see edges and vertices which are behind the model.

Curved Poly Editor can also show backgrounds grids which support the modeling of elements in scene. This backgrounds grids are visible when **grids** is selected. It is also possible to activate snap to grids to make any transform operation **snap to grids points**.

<< Back To Index

Parts List



Curved Poly Inspector

Parts List

●	MokaBase	S	Metal_
○	MokaCap	S	Metal_
○	MokaCapScrew	S	Metal_
○	MokaCupHolder	S	Metal_
○	MokaHandle	S	Metal_
○	MokaHoldingStructure	S	Metal_
○	MokaUp	S	Metal_
○	HoldingRing	S	Metal_

Fig. 46 Parts List

Another common tool, which is accessible from almost all the Tools Set, is the **Parts List**. Each Curved Poly model can be split into more parts, each part becoming a distinct sub mesh once the model is tessellated into more meshes. With parts list we can rename the parts, or change their material. You can also addictively select each part polygons with the **s button** at once. Each part added to selection this way has its name highlithed in the parts list.

Furthermore. At any time, there is a part which is active. And you can control which one it is here in the parts list, using the radio button on the left of a part name. There are tools in Curved Poly Editor which need to know on which part they have to operate, so it is a good thing to keep. An alternative way to activate a part is using **Parts Mode**. With parts mode we can click on a part to, not only select it, but also make it active.

<< Back To Index

Shape Tools

You can find more about shape tools on the **full documentation page**, or you can watch the **Shape Tools Video on youtube**

Shape Tools is the most important Tools Set in Curved Poly, allowing you to change the shape of every Curved Poly model you want to start with. We are going to discuss every aspect of it into 5 specific sections:

- **Selection Transform**: the group of transforms applied to any selection of the model, exploiting both gizmos in scene and the Dedicated Inspector.
- **Backgrounds**: the combination of background images and grids can be used to support you while modeling.
- **Handles and Edges**: handles and edges are very important to control the shape of your model.
- **Polygons**: Curved Poly works with algorithms which fills the countour of curved edges a curved polygon is made of with a proper set of real polygons. This section is about the way this algorithms can be set on the polygons.
- **Shaping Options**: a further set of shaping options are available, controlling how unselected parts of the model react to **Selection Transform**.

<< Back To Index

Selection Transform

Let's start working with Shape Tools. We have imported a sample model, and maybe we can spend some time to assign to its part some material before proceeding. This can be done by dragging the material on the model, or in the parts list.

Now, according to our selection mode and selection tool, we can select one or more elements on the model. Now a transform gizmo appears at the center of our selection. It can be used to move it ... or rotate it ... or scale it, according to the unity tool which is active.

In the Dedicated Inspector here you can find a Selection Transform tool, which is used to keep control over transforms parameters. It works pretty much as Unity Game Objects transforms. The interesting part here is the Edit Pivot section, where you can fix the reference system you transform is using. By activating the Edit Pivot Mode, we can change it's position. If we change the position of the pivot and we switch back to transform mode, we can see that the selection is rotated or scaled with respect to the new pivot position.

During modeling, you may need to control the position of your selection with some precision. For instance, let's select an edge. Let's also move the gizmos at the position of one of its vertices. Now we can move the edge by dragging around that vertex. Let's now activate Vertex snapping on the x component. The default threshold will be fine. When we the the edge, the x component of it's pivot, which is its first vertex here, will shap to the x position of any other vertex. A red line is rendered here, showing that the snapping is happening.

<< Back To Index

Backgrounds

A better way to position transform and gizmos is using backgrounds. **Backgrounds** in Curved poly editor are flexible elements, which can be used both as image references and as grids. Let's say they are background grids which can be assigned an image. We can add a backogrund with **Add Background**. Then we need to set it up. We can place it where we want using its position or extension gizmos. We can also control such positions in the Bacground Grid section of the Dedicated Inspector, which appears only if you cklick on one of the Background gizmos. Here you can also add an image, and control the number of nodes in the grid, and their properties.

Once your grid is set up, you can snap to it if Snap to Grids is active. The snapping happens on the edit pivot of the selection transform.

If you don't want this to happen at some point, you can always disable the snap to grids options, or even the **grids visibility** to hide them.

Since placing grids may take time, it is useful to save them, using **Save Backgrounds** to generate a backgrounds asset which contains them.

Backgrounds assets can also be edited in Unity Inspector, once generated. Stored Curved Poly Backgrounds can be reopened on need, while editing other shapes, with **Load Backgrounds**. Curved Poly Editor already comes with a little set of premade Backgrounds which can be usefull. You can load one, and then you can decide to **Remove** the Backgrounds you don't need, or you can also **Remove all Backgrounds at Once**. Once you have a background picked in scene, you can also align the Scene View Camera to it, with **Align Camera**, so that is gets behind as intended.

<< Back To Index

Handles and Edges

Handles are used to give shape to edges. Each curved edge has two vertices, and for each vertex it has two handles. Handles are not directly shown on the model, because you need to pick first a vertex or an edge to have access to their related handles.

So in order to pick and handle, we need first to have the handles visibility option active. We also need to be either in Vertex Mode or Edges Mode. For instance, if we are in Vertex Mode we can pick a vertex first and then we can pick one of its related handles.

A picked handle is rendered as a set of 3 arrows. On the dedicated inspector a section appears about the handle, called **Picked Handle**. From this section we can pick back the vertex attached to the handle with **Back to Vertex**. So, pick again the handle. We can also pick the edge it belongs with **Back to Edge**. Handles are visible only when their edges or vertices are picked, so you need to use the **Pick Tool** to access them. Back to vertex and Back to Edges can be used to navigate the model, and to investigate the exact relationships between elements in the model.

There are two kind of handles in Curved Poly: **free handles** and **constrained handles**. We will start with constrained handles, since they are the most common. A constrained handle has a 🟥 magenta, 🟨 yellow and 🟦 cyan arrows where the yellow and cyan arrows have a circular shape.

Such handles are constrained, as all the handles on their vertices, because that vertices are smooth. In order to keep the smoothness, all the handles must lay on the same plane. This plane is defined by a vertex editing normal, which is drawn on top of the vertex. So, about the three handles: the 🟥 magenta handle controls the distance from the vertex. When you use this, the handle keeps constrained to the plane, so the length handle doesn't affect the other handles. The 🟨 yellow arrow is called the roll of the handle. It controls how much the handle rotates around the editing normal. So, again, when you change the roll, the handles keep being on the same plane, and its position doens't affect the smoothness plane, neither the editing normal on the vertex, or the other handles. The cyan arrow is the pitch. It makes the handle rotate in a way which force the whole editing plane to rotate together with the handle, in this sense affecting both the editing normal on the vertex, and the other handles.

Handles have also a **Weight parameter**. This parameters affects the shape of the curve without changing the actual handle. If you are familiar with weighted curves, from other modeling tools, you should how it works. If you are not, let's say that the more the weight is high, the more the curves is dragged towards the handle. The less is the weight, the less the curve is attracted by the handle. You may try it a bit on your own to understand this.

Below the parameters controls, the Dedicated Inspector expose this actions:

- Back to Vertex , this Button to activate the vertex this handle refer to. It is useful in situations in which you have much stuff in the scene, or too many controls on the object, and you find hard to access the vertex.
- Back to Edge , Press this Button to activate the edge this handle belongs to. It is useful in situations in which you have much stuff in the scene, or too many controls on the object.
- Center Handle , Correct the Roll Value of the Handle, so that it becomes centered with respect of the two other handles it has on its left and right. If the two handles are aligned, the centered handle will have an exact 90° angle with both them.
- X-Y-Z Handle Directions , correct the direction of the handle by aligning it to the x, y, z directions. Left-Right (X Button), Up-Down (Y Button), Backward-Forward (Z Button) directions are respectively chosen depending on the current direction. Such correction will affect only the Roll Value. Note: the

> software will keep always a 90° angle with the Editing Normal during this changes in order to keep the whole surface smooth. If the x,y or z directions are not on 90° with the normal, the handle direction will be correct to the closest direction available satisfying such constraint.

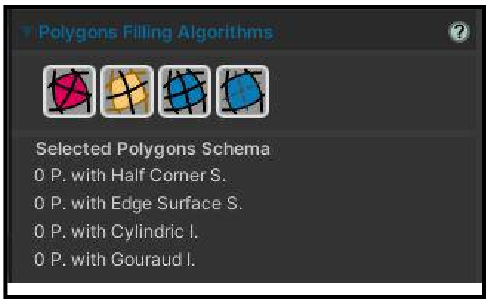
Thery often, two handles are aligned together. This alignments are part of the structure of the handles, and can be added or removed only with Editing Tools, in the process of making custom primitives. When you are in Shape Tools, you can see if two handles are aligned by looking for a blue handle. For instance here we have picked an handle, and on the same vertex, opposite to it, there is a blue handle. The blue handle is aligned with the one we have picked. An aligned handle is affected not only by the pitch of the picked handles ... but also by the roll. So they will rotate together around the vertex, keeping aligned one with the other. Again, the length arrow and the weight control do not affect the aligned handle.

In the inspector you can find also some more tools to control the handles direction. The direction of the handle is the x, y, and z, of the vector going from the vertex to the handle. You can directly assign it here, but be aware that using the direction controls will affect the roll, pitch and length at once in more complex ways. One useful instrument here is the Center Handle. With the center handle you can force the roll in way that makes the handle centered, with respect to the two other handles on the left and on the right. If the two handles on the left and on the right are aligned, this will surely put the picked handle at 90 degrees with both of them.

Now, let's talk about free handles. One possible situation in which handles are free is when a vertex is marked as sharp. Sharp vertices are the ones represented with an empty square. You can decide which vertices are sharp with the Editing Tools. Handles of sharp edges are not constrained. Here the editing normal on the vertex is less relevant, still important though for a few processes used in other tools sets. For free handles the dedicated inspector looks the same, but the roll and pitch arrows are not in scene. Here we are rather free to change the direction of the handle with a simple move tool. Of course, it is intended, when you change a free handle, sharp features mat appear on the model or not, depending on how you place the handle.

<< Back To Index

Polygons



Selected Polygons Schema
0 P. with Half Corner S.
0 P. with Edge Surface S.
0 P. with Cylindric I.
0 P. with Gouraud I.

Fig. 61 Polygons Filling Algorithms

Let's say we have selected at least one polygon. The dedicated inspector may have a section called **Polygons Filling Algorithm**. This depends upon the Selection Mode in use: to have the section you need to be either in **Polygons Mode**, or **Parts Mode** or **Rings Mode**. This sections controls how real polygons are generated inside curved polygons.

There are more ways Curved Poly can use to **fill** a curved polygon with a proper set of raw polygons (i.e. triangles). Each way is a specific procedure we call **Filling Algorithm** (or also **Interpolation Algorithm** or **Interpolation Schema**). Since they change the way each specific Curved Polygon appears, it's possible to use different **Filling Algorithm** on different curved polygons. You can select any group of Curved Polygons here, and click on one of the buttons to try out a different procedure for each curved polygon which doesn't look the way you want it to look.

How to Use this: you can select any group of Curved Polygons here, and click on one of the buttons to try out a different **Filling Algorithm** Tip: there is a lot of work around Filling Algorithms at MushroomsLabs. We literally researching on more and more alternatives to do this. You may find new alternative algorithms in newer version of Curved Poly, each one improving the way polygons are shaped. Version 1.2 introduced a new algorithm called **Half Corner Surface**, we are planning to introduce a better one we are working on in one of the 1.3.x

Half-Corner Schema: this is a very complex interpolation schema, generating wonderfull polygons in most of the cases. Since version 1.3, it's the default choice for every new model generated with Curved Poly Maker. It's not perfect, but slightly better than **Edge Surface Schema**, and slightly slower. In most of the cases, you should not see a bit difference, but if something looks wrong with this schema, you should switch to one of the others.

Edge Surface Schema: this is a very complex interpolation schema, generating wonderfull polygons in most of the case. If the edges pattern is too complex, this may generate some strange behaviour.

Cylindric Interpolation: this is a fast one and works perfectly in situations where edges across vertices satisfy some specific constraints (you need to have 4 edges on a corner, and opposite handles must be aligned; this is the case of almost all primitives of Curved Poly - Shape Editor). *Cylindric Interpolation ignores Edges Editing Normals*. Also, this is the reccommended algorithm in case you have very little polygons or polygons with a very low Level of Details or low Tessellation Hints.

Gouraud Interpolation: same as **Cylindric Interpolation**, but the way normals are computed is more simple

<< Back To Index

Shaping Options

We have discussed almost everything about shaping. There is left a section here called **Shaping Options**. This section is responsible of a few important options which control how the model will react to shaping actions.

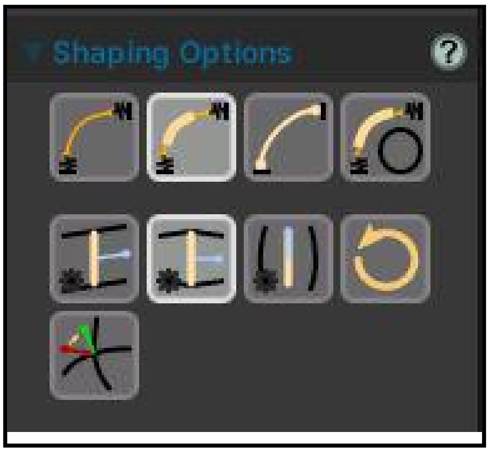


Fig. 62 All Shaping Options

Elasticities

Let's start with elasticities. If you pick a vertex and move it, the edges attached to it change shape. Let's look at this edge for instance. This happens because one side of the edge is changing, while the other side of the edge is not. This is pretty much common in graphics softwares which use curves. Here in Curved Poly, you can ask the editor to do something about this. Elasticities indeed controls the way in which edges reacts to reshaping actions.

The most simple option, but not the most reccomended, here, is **Clayish** mode. In Clayish mode, the edge does not react to shaping at all. This is waht happens in most of the other graphics tools around, so we think you should used to it.

The other elasticities works in this way: they change the lenght of the handles of the edge, to match some shaping action. So if I move this vertex, with this handle picked, the handle change its length during the process.

So, with the **Proportional** mode, active by default, if the lenght of an edge decreases, also the length of the handles decreases. Proportional mode follow the principle of keeping the shape of the edge the way it is, under some transforms, by scaling the edge proportionally.

The opposite to Proportional is **Elastic**. In elastic mode, when an edge length would increase, the handles react like springs reducing their length, and if the edge length decreases, the handles react by increasing their length. The principle here is to make the edge work like a spring indeed, making its shape change drastically to match as much as possible the original length.

Last, but not least, we have the **Elliptic** elasticity. This works works in a way which is similar to proportional, but it stretches the shape of the edges such that elliptic arcs keep being elliptic arcs, so it works very well on shapes which has been designed to have circles or ellipses.

Furthermore, while you are shaping a group of elements, you can try out the elasticities and choose the one which looks better for you. To do so, you just transform your selection a bit, then you go to Shaping Options and change the elasticity mode. This will recompute the transform on the selection, updating the edges which are affected by elasticities settings.

Edge Normal

Another important group of tools here, are the ones which control editing normals on groups of elements. We have already covered the role of editing normals, both on vertices and on edges. As you can see here, we have three options which are similar to the ones used for edges normal. The difference is that, like elasticities, these options applies by default on all edges which are changing shape. Again, by default the option here is the **Axis Normal**, which is usually the best solution, but you could try to work with the others.

Each curved edge has as a Normal vector assigned to its center. This normal may be used by the Interpolation/Filling Algorithms to correct the orientation of polygons across the edge, so it's very important that you understand what you can do to correct its orientation.

When an edge is reshaped, its normal must be recomputed. Here are the algorithms which are automatically applied during the process.

Axis Normal (Default-Optimal Choice): an algorithm which takes into accounts the shape of the Edge, the two editing normals on the two extremes vertices and also the direction of other edges nearby the extremes. Such direction is called the Axis. Axis Normal is more correct to use in situation where there are sharp angles between edges.

Edge Normal: an algorithm which takes into accounts only the shape of the Edge and the two editing normals on the two extremes vertices.

Balance Normal: an algorithm which takes into account the shape of the Edge, the two editing normals on the two extremes vertices and also the position of any opposite edge (for quads) or opposite vertex (for triangles) on adjacent polygons. The Editing Normal will be perfectly balanced between the two opposite positions: this is correct on spherical surfaces equally subdivided (as it is for spherical primitives) .

Reset Normals: reset edges normals on all selected edges.

Fix Normals: recompute all the normals, both the ones on vertices and then the ones on edges. Best used when something looks weird in the way edges normals are assigned.

<< Back To Index

Tessellation

> You can find more about tessellation Tools on the **full documentation page**, or you can watch the **Shape Tools Video on youtube**

Curved Polys tessellation into real polygons is all about determining how much segments should be used to tessellate each curved edge. Then, the **Filling Algorithms** on each polygon will do the rest.

Tessellations Tools is dedicated to the setup of **Tessellation Hints** and other tessellation settings. Tessellation Hints are little labels (all the characters from **A** to **P**, in alphabetical order) which describe how much segments should be used to tessellate an edge.

> Since we are working deeply on the way curved polygons are tessellated into real polygons here, it is a good idea to do something to visualize a wireframe of such polygons. In unity this can be done in the Scene Toolbar either from **Gizmos>Selection Wire**, or changing the

Shading Mode from Shaded to **Wireframe** or **Shaded Wireframe**.

The exact number of segments used on each curved edge, is usually defined at Run-Time (or during shaping or editing processes), after a specific Level of Details as been assigned to the entire model; such Levels of Details are defined inside LoDs tables, which are stored in **LoDs Assets**. Usually, **A** means a very little amount of segments (the minimum being 1) and **P** means an impressive huge amount of segments. For practical purposes, We suggest to use only the hints from **B** to **G**, reserving the **A** hint for special situations in which you only need to have a **Placeholder*** , and all the Hints after **G** for rare situations or special uses.

If you need to check the impact of editing on more Levels of Details at once, you should use the **LoDs preview Window**. This can be opened from the Curved Poly Component. Once this window is open, it will show changes happening on your models at different resolutions. You will also see an overview of the total amount of vertices and triangles used by each version of the model.

Furthermore, for a complete interaction with the model, we have added to Tessellation Tools a control over **Polygons Filling Algorithms**, not different from the one used in Shape Tools. With this control you can correct potential issues generated on polygons after the changes made to hints and level of details, and also try different combination of filling algorithms with tessellation data, in case you need to lower the computation time of tessellation.

<< Back To Index

Levels of Details Table Assets

Curved Poly tessellation algorithms are based on a Levels of Detail Table which can be customized. Such table is stored in the project with a specific asset called LoDs. You can find the default settings for the LoDs table at **Assets/CurvedPoly/LoDs/LoDs**. If you open the asset, it should look like this:

Each LoD is a record in the table and you can add how much LoDs you wish with the **Add LoD** button. On each LoD you will find a set of sixteen integers value from **Hint A** to **Hint P**. Each hint contains the amount of segments an edges should be tessellated with. We are going to see how hints are assigned to edges later.

By default, you will find six clearly understandable LoDs named **Close, Near, Normal, Distant, Far** and **Box**. Their values have been set up to be coherent with our tessellation algorithms. That said, for practical situations you will probably choose to use the default settings and limit the use of LoDs only to some values, maybe Near-Normal-Far which is a fair enough choice.

If you wish to change LoDs records you can:

- Create your own custom **LoDs** asset in the **Project** Panel popup menu or with the Assets Menu selecting **Create > Curved Poly > LoDs**.
- Add new LoDs with the **Add LoD** button. Pressing more times the Add LoD will automatically create a set of incremental LoDs values which may be a good starting point.
- Clone a pre-existent LoD with the **Clone** Button inside the LoD record.
- Clone a pre-existent LoD with doubled values with the **Clone(2x)** button inside the LoD record.
- Remove any load you don’t wish to use with the **Remove** button.
- Rename the LoD the way you like. You should always use unique names to make it simple to distinguish them.

Below the records of the table you will also find the helpful hint views, which contain the same data organized by hint (the column of the table) and not by LoD (the row of the table).

When you have a **LoDs Asset** (either the default one or a custom one), you can assign it on your **Curved Poly Behaviour**.

The choice above the LoDs Asset will be automatically updated with the list of all available LoDs. Each time you change the LoD, your **Mesh** will be recalculated with the alternative Tessellation Values. This Mechanic is available both in edit mode and in play mode, since its part of **Curved Poly Runtime**.

<< Back To Index

Levels of Details

When tessellation tools are active, Curved Poly will show tessellation hints on all selected edges. Then, you can use any of the Selection mode we have previously covered to select the edges you want to change the hints on.

On each edge there are **two hints**, one for each side of the edge. If you assign the same hints on both the side of the edge, curved poly will tessellate that edge with the exact amount of segments which are assigned in the LoDs table. If the hints are not the same, Curved Poly computes a continuos gradient of segments, where the segments are smaller where the hint is high, and larger where the hint is low.

<< Back To Index

Composition Tools

Composition Tools are thought to help you rearrange different parts from different models into new models.

- The **Parts Tools** is used to setup an object parts. Parts can indeed be rearranged and generated from scratch the way you need.
- The main **Composition Tools** section is designed to help you in joining and splitting parts from different models.
- The **Create Primitive** section allows the instancing of Custom primitives through well defined parameters.

Composition Worflow Example

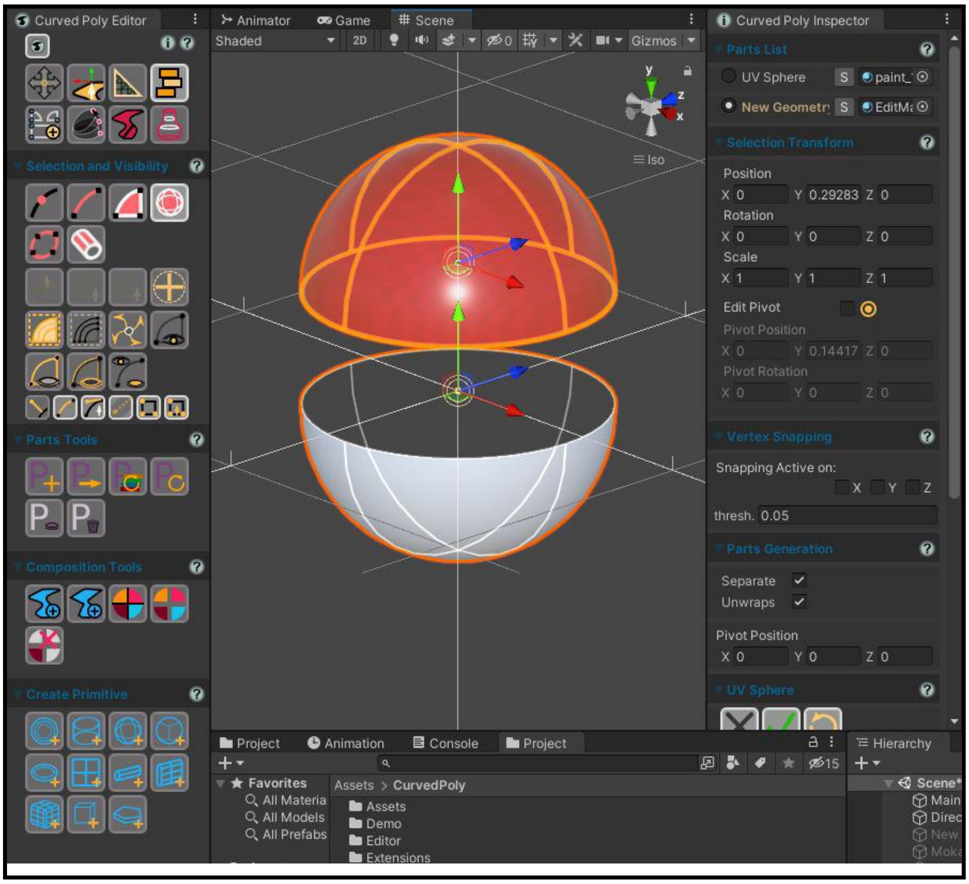
Let's see a simple workflow which exploits all the sections of Composition Tools.

We can generate an empty Curved Poly using **Create New** in GameObjects Tools. Then we can come here to composition Tools and Create e new **Sphere UV**.

Let's be in **Vertices Selection Mode** and select the topmost vertex, then press **Grow** to extend the selection to the top half part of the sphere.

We can now generate a new part with **Generate Part with Selected Polygons**. Now the upper part is using a new default material, but we can assign the materials we wish to use in **Parts List**. The parts here are still connected, so they have vertices and edges in common, and if you move one of the common vertices both the two parts will be reshaped.

Finally, we can call for **Separate and Explode** to split the model into two independent parts. The two parts are now completely separated and they can be used and rearranged with other parts generated on other objects.



Position
X 0 Y 0.29283 Z 0
Rotation
X 0 Y 0 Z 0
Scale
X 1 Y 1 Z 1
Edit Pivot
Pivot Position
X 0 Y 0.14417 Z 0
Pivot Rotation
X 0 Y 0 Z 0
Snapping Active on: X Y Z
thresh. 0.05
Separate ☑
Unwraps ☑
Pivot Position
X 0 Y 0 Z 0

Fig. 79 Separate and Explode applied on the Moka Premade Model. After the model has been exploded, you can go on deleting, rearranging and moving parts to compose a new model.

<< Back To Index

Primitives

With Composition Tools it's possible to generate new custom primitives for your model. There are a few important settings which are controlled in the dedicated inspector:

- **Separate (Geometries)**: each part will be a different Geometry. When disabled, all generated parts will be added to the same geometry. For this reasons, this operators also shows the **Geometries Inspector** which will be presented in the **Geometries Operator section**
- **Unwraps**: where possible, parts generated will have a default pre-unwrapped set of uv-coordinates. Avoid this if you plan to edit geometries with the **Edit Operator** (pre unwrapped models don't work well there). In any case, you will be able to add unwraps at a second time with **Materials Tools**.
- **Pivot Position**: position where primitives will be placed when generated. There is also a gizmo in scene to control this.

Below the Parts Generation section, new sections will appear depending on the primitive which you are generating.

The **Create Primitive** section in the Main Editor Window has a group of buttons, each one generating a different primitive:

- **Circles**
- **Cylinder**
- **UV Sphere**
- **Quads Sphere**
- **Donut (Torus)**
- **Quads**
- **Plate**
- **Bar**
- **Chamfer Box**
- **Cube**
- **NGon**

Once a primitive is generated, a primitive builder is activated. This builder generate the primitive with default parameters first, then allows you the customization of the primitive parameters in order to build them again. Each primitive builder has its own parameters. Common actions for all the builders are:

- **Apply**: Confirm any change and exit the builder (You can always undo this if you change your mind). Pressing **Enter** will do the same.
- **Close**: Remove any built content from the model and exit the builder (You can always undo this if you change your mind). Pressing **Esc** will do the same.
- **Update**: changes made to the primitive parameters are not instant. You need to press this to rebuild the primitive with different parameters.
- **Reset**: reset the primitive to its default parameters values and update it.

> If you change primitive or even Tools Set while one of the builders is active, the operation will be automatically applied and the builder will be closed.

Last but not least, remember that primitives are generated at the position of the **Pivot**, which is drawn in the scene view as a target with 2 circle and one central sphere. clicking the target will allow you to move the pivot, or you can write the pivot position from the menu control. After the pivot has been moved, you need an **Update** to place the primitive at the new position.

Circles

The Circles Builder builds a set of filled concentric circles. There is no options to have an empty circle, but you can always create a filled one and switch to the **Edit Tools** to cut the interior. Parameters are:

- **Radius (Main Circle)**: radius of the most external circle

- **Size**: number of concentric circles
- **Sides**: number of edges each circle will be made of.
- **Main (Tessellation)**: the hint assigned to the circles edges.
- **Sub (Tessellation)**: the hint assigned to edges used to connect circles.

Cylinder

The Cylinder Builder builds a Cylinder (or a Capsule, depending on the parameters assignment). Parameters are:

- **Radius**: radius of lateral area of the cylinder.
- **Height**: height of lateral area of the cylinder.
- **CapH**: height of the caps extensions to the top and bottom of the cylinder.
- **Size**: number of strips in the cylinder lateral surface. Each strip is made of a closed set of quads limited by two circles, and the number of circle is always Size+1.
- **Sides**: number of edges each circle will be made of.
- **Main (Tessellation)**: the hint assigned to the circles edges.
- **Sub (Tessellation)**: the hint assigned to edges used to connect circles.
- **Caps (Tessellation)**: the hint assigned to edges used in the caps.
- **Top (Part)**: when disabled, the top cap is not built.
- **Bottom (Part)**: when disabled, the bottom cap is not built.

UV Sphere

The UV Sphere Builder builds a Sphere by connecting a set of parallel circles. Parameters are:

- **Radius**: radius of the sphere.
- **Size**: number of strips in the sphere. Each strip is made of a closed set of quads limited by two circles, apart

from the first and last strip which are made of triangles, being limited on one side by a single vertex.

- **Sides**: number of edges each circle will be made of.
- **Main (Tessellation)**: the hint assigned to the circles edges.
- **Sub (Tessellation)**: the hint assigned to edges used to connect circles.
- **Bottom (Part)**: when disabled, the bottom half of the sphere is not built.
- **Left (Part)**: when disabled, if also Bottom is disabled, the left half of the sphere will not be built .
- **Back (Part)**: when disabled, if also Bottom and Left are disabled, the back half of the sphere will not be built .

Quads Sphere

The Quads Sphere Builder builds a sphere made only of Quads. This one has few parameters:

- **Radius**: radius of the sphere.
- **Main (Tessellation)**: the hint assigned to the circles edges.

Donut (Torus)

The Donut Builder builds a Donut (Torus) by connecting a set of circles. Parameters are:

- **Radius**: radius of the main Circle in the Donut.
- **Section**: radius of the section of the Donut.
- **Size**: number of strips in the donut. Each strip is made of a closed set of quads limited by two circles, the number of circles is also equal to size.
- **Sides**: number of edges each circle will be made of.
- **Main (Tessellation)**: the hint assigned to the circles edges.
- **Sub (Tessellation)**: the hint assigned to edges used to connect circles.

- **Bottom (Part)**: when disabled, the bottom half of the donut is not built.
- **Outside (Part)**: when disabled, the circles and polygons in the external side of the donut are not built.
- **Inside (Part)**: when disabled, the circles and polygons in the internal side of the donut are not built .

Quads

The Quads Builder builds a Grid of Quads. Parameters are:

- **Width**: width of the grid.
- **Height**: height of the grid.
- **X**: number of rows in the grid.
- **Y**: number of columns in the grid.
- **Main (Tessellation)**: the hint assigned to all the edges.

Bar

The Bar Builder builds a geometry which has the same structure of a cylinder, but different parameters and different setup,in order to make the primitive look like a bar. Parameters are:

- **Width**: width of the bar.
- **Height**: height of the bar.
- **Depth**: depth of the bar.
- **XYZ ext (Bevel)**: length of the smooth caps at the beginning and end of the bar.
- **UV ext (Bevel)**: modulate uv coordinates generation on the smooth caps at the beginning and end of the bar.
- **Size**: number of strips in the bar. Each strip is limited by two ellipses.
- **Main (Tessellation)**: the hint assigned to all the edges.
- **Bevel (Tessellation)**: the hint assigned to the edges used in the caps.
- **Bisided (Parts)**: when disable, the back half of the bar is not built.

Plate

The Plate Builder builds a Plate. A Plate is like a grid of Quads with some thickness (depth) smooth bevels on the sides. Parameters are:

- **Width**: width of the plate.
- **Height**: height of the plate.
- **Depth**: depth of the plate.
- **XYZ ext (Bevel)**: length of the bevels.
- **UV ext (Bevel)**: modulate uv coordinates generation on the bevels.
- **XSize**: number of rows in the grid.
- **YSize**: number of cols in the grid.
- **Main (Tessellation)**: the hint assigned to all the edges of the grids.
- **Sub (Tessellation)**: the hint assigned to all the edges of bevels.
- **Bisided (Parts)**: when disable, the back half of the plate is not built.

Chamfer Box

The Chamfer Box Builder builds a Chamfer Box. Parameters are:

- **Width**: width of the box.
- **Height**: height of the box.
- **Depth**: depth of the box.
- **XYZ ext (Bevel)**: length of the bevels.
- **UV ext (Bevel)**: modulate uv coordinates generation on the bevels.
- **X (Sides)**: number of internal separations on X direction.
- **Y (Sides)**: number of internal separations on Y direction.
- **Z (Sides)**: number of internal separations on Z direction.
- **Main (Tessellation)**: the hint assigned to all the edges of the grids.
- **Bevel (Tessellation)**: the hint assigned to all the edges of bevels.

- **Top (Parts)**: when disable, the top face of the box is not built.
- **Bottom (Parts)**: when disable, the bottom face of the box is not built.
- **Right (Parts)**: when disable, the right face of the box is not built.

Cube

The Cube Builder builds a Cube made of linear edges and sharp vertices. It has no parameters

NGon Plate

The NGon Builder builds a Plate which has as base a polygon of a custom number of sides. Parameters are:

- **N**: the number of sides. So for example 5 generates a pentagon, 6 generates and hexagon.
- **Radius**: radius of the circle used to build the vertices of the custom polygon.
- **Height**: height (thickness) of the plate.

<< Back To Index

Custom Parts

With Composition Tools it is possible to split a Curved Poly model into separated parts, where each part can be assigned a different materials (as we have already seen in the **Parts List**). Each part contains a different subset of the model polygons, and **will become a submesh** in the final computed mesh once any interpolation process is applied.

the **Parts** section of the Tools Set has 6 available actions:

- **Generate Part with Selection**: Generates a new part using selected polygons.
- **Move Polygons to Part**: Removes the selected polygons from their parts and put them in the actual selected part. You can change the selected geometry using the parts inspector which will be explained in the next subsection.
- **Reset Parts**: Clear the model from separated parts, merging back all the polygons into one big part.
- **Reset Parts by Mat**: Merge together into one single part all the parts using the same material. The final amount of parts will be the exact amount of materials which has been assigned in the inspector or in the Parts List.
- **Hide Unselected Parts**: Hide all the parts but the one being selected. You can change the selected part using the parts inspector which will be explained in the next subsection.
- **Destroy Unused Parts**: Destroy any empty part. Parts may get empty after after its polygons have been cancelled in the Edit Operator, or after they have been moved into other parts.

Newly generated parts will automatically be assigned a new material, taken (by default) from a pool of materials which can be configured in the settings: check the **Setup** section for this.

Advanced Parts Inspector

The Advanced Parts inspector is a window which gives you details about your parts. It can be opened with the menu at **Tools>Curved Poly>Others>Parts Inspector**. For each Part the inspector renders a foldout panel which can be opened to have more informations and controls over each part. From the inspector you can change or view:

- the **Active Part**: only one part can be active at a time. The active part will be used in many other operators, like the **Edit Tools** or the **Composition Tools** when they have to generate new polygons. You can change the selected one by clicking on the checkbox near its name, which is accessible reguardless the part inspector foldout being open or closed.
- **Name**: It's not mandatory to have named parts, but it's useful. The name can only be changed when the foldout is open, but it's still visible when it's closed.
- **Polygons Count**: shows the amount of polygons in the part.
- **Select Polygons**: select all the polygons in the part at once.
- **Unwrap Groups**: the list of unwrap groups in the model. You have the option here to select all the polygons belonging to an unwrap group at once. See **Materials Tools** for more details about unwrap functions.
- **Unwrap Functions**: unwrap functions are functions used to generate unwrap groups. See **Materials Tools** for more details about unwrap functions.

<< Back To Index

Compositing

Use this to generate new models mixing parts from existing models.

Import: Insert a copy of any **Curved Poly Asset** in this model, adding its parts as separated parts. This parts can be reworked with the other tools here.

Import Mesh: Insert a copy of any **Mesh Asset** in this model, changing the Mesh into a Curved Poly made of lines and sharp edges, adding its parts as separated parts. This parts can be reworked with the other tools here. The linear edges and the sharp vertices can be furtherly transformed into smooth parts using **Edit Tools**

Separate: splits a model into its part. This is useful when you have a model whose parts has some shared edges. The shared edges are duplicated, so that different parts can be moved freely from eachother.

Separate and Explode: works like separate, but also move away each part from the center of the model, in this way it's easier to identify each separated parts.

Tip: you can use generate parts to create new parts, and then separate the parts here. This has the effect of cutting a model into more little pieces, which can than be combined together into more complex objects.

Delete Part: delete a part and its polygons from the model.

<< Back To Index

Materials Tools

Every time we work on a model, we also need to check how materials will applied on it. Even if Curved Poly Assets and models are thought to contain geometries which can be setup with different materials, no material can be properly assigned without assigning some uv on the model

The **parts generation section** has already covered in details how we can split a Curved Poly model into more parts, each one with a different material, and each one converted into a single submesh to be rendered by Unity.

Once of the most important aspect of **parts** is that each part is also arranged into more **Unwrap Groups**. An **Unwrap Groups** is a subset of a part polygons which has its own UV assignment. While the polygons may be attached in 3D space, the **Unwrap Groups** maybe separated in UV space, and can be edited one by one

Unwrap Groups are generated using **Unwrap Functions**. This functions also assign good uv, so they are to be considered a fast way to generate UVs, or at least a good starting point. After unwrap groups have been generated, it is possibile to customize the uv coordinates generated using the un panel.

- The **Unwrap Tools** section covers in details how Unwrap Groups are generated
- The **UV Panel** section covers in details how UVs can be customized using the UV Panel.

<< Back To Index

Unwraps Tools

Unwrap Functions are tools which can be used to automatically generate uv coordinates (unwraps) for your needs. The general way to use the tool should always follow this steps.

- **(A)Activate a Part** : you can do this with the Parts List, or with the parts inspector, or even using Parts Selection Mode and clicking directly on the part you wish to activate.
- **(B)Select a Group of Polygons** : you can do this using the common tools, but be aware that **only polygons belonging to the actually active part will be affected**. You can still select other polygons using actions like

Select All or **Grow**, but the tool will only affect selected polygons belonging to the active part.

- **(C)Choose an Unwrap Function** : when you select a function, an instance of it will be added. The function will generate one or more unwrap groups, using the polygons you have selected in a way which depends on the function.
- **(D)Fix the Unwrap Function parameters** : an Unwrap Function has many parameters, which can be controlled after you have generated it. You can find them in a dedicated section in the Dedicated Inspector. You may change and recall (reset) the Unwrap Function and its parameters more times before achieving a good unwrap or an unwrap which can be a good starting point before moving to the **UV Panel**.

While making Unwrap Groups with Unwrap Functions, you should always keep in mind that:

- A Part can have one or more Unwrap Groups.
- A Part can have one or more Unwrap Functions.
- An Unwrap Function can generate one or more more Unwrap Groups. The total of the Unwrap Groups in the Part is the sum of Unwrap Groups generated by its functions.

So: there is a hierarchy involving Parts, U.Groups and U.Functions. You may find it a bit complex at the beginning, but you will find it pretty practical once you have used it a few times, especially because Unwrap Functions can be assigned and modified only in **Materials Tools Set** and they don't affect any other Tools Set. They exist only to make it easier the generation of Unwraps Groups. The key point here is that you can have more Functions for the same Part so that you can cut the part into different subparts to apply a different unwrap function to each subpart, and this is very important when you need to unwrap complex parts made of a lot of polygons and with irregular shapes.

There are 6 available unwrap functions:

- **Planar Unwrap (Bisided)**: Generates uv coordinates mapping 3D coordinates on a plane. It is bisided, meaning that by default it will generate 2 Unwrap Groups, splitting the model into two halves watching the two sides of the plane. You can set it to 4 or 6 sides, to get 4 or 6 Unwrap Groups generated by cutting the model with respectively 2 or 3 planes.
- **Cylinder Unwrap**: Generates uv coordinates mapping 3D coordinates on a cylinder. By default it will generate 2 Unwrap Groups, splitting the model into two halves watching the two sides of a reference plane containing the axis of the cylinder.
- **Spherical Unwrap**: Generates uv coordinates mapping 3D coordinates on a sphere. By default it will generate 2 Unwrap Groups, splitting the model into two halves watching the two sides of a reference plane passing through the center of the sphere.
- **Physical Unwrap**: **This is an experimental function**. It is an improved version of Planar Unwrap Bisided which takes into account the arc length of edges in 3D space to improve the edge shape in uv space.
- **Edge Length Unwrap**: is is planned to become a better version of physicalunwrap, but it's still experimental. This does its best to keep a proportion among the length of edges.
- **Empty Unwrap**: An Unwrap which doesn't assigne values, generating only one Unwrap Group. It's the default function for imported parts or parts generated with the **Create Operator**. If you press the button, you will get an unwrap with all uv coordinates assigned to 0 (zero).

Unwrap Functions Parameters

The most important parameters of a Function are its center and reference directions, which are used to control cutting planes. This parameters can be edited directly on a set of Handles which appears in the scene view: a white sphere is the center; a red line ending with a sphere is the first direction, and a green line ending with a sphere is the second direction. You can point on a sphere to access a **Move Tool** to change its position. Around the frame made by the two directions you will see 2 (or more) yellow quads showing the sides of the unwrap functions, that is the reference used during the generation of each unwrap group. These parameters can be changed also with a menu. All these controls are available only after a Function has been generated or selected (to select a function, you need to look for it in the in-scene Parts Inspector on the right).

Other parameters are:

- **Physical Size**: this parameter controls the rapport between distances in 3D space and distances in uv space. This rapport can't be applied perfectly on unwrapped curved surfaces (this is a classical problem of unwrapping), but each unwrap function will work to do its best to get close to it. **Physical Unwrap** has been designed to match **Physical Size** better than the others.
- **Sides**: this parameter controls amount of Unwrap Groups generated by the function.
- **Overlap**: this parameter controls the way the unwrap groups are placed in uv space If true, the groups will be overlapped, otherwise they will be placed at different position in uv-space.
- **Margin**: this parameter is available only when overlap is false. It controls the spacing between adjacent groups placed at different position in uv space.

Most of these parameters will not affect directly the function until you Reset the Unwrap with one of this actions:

- **Click on any Unwrap Function**: You can change the unwrap function. The new function will keep your parameters assignment, then will recompute the Unwrap.

- **Reset Unwrap**: recompute the unwrap taking into account any updated parameter in the Function.
- **Unwrap Frame on XY**: set the Unwrap Frame so that its directions become the x and y axis, than it recomputes the unwrap.
- **Unwrap Frame on XZ**: set the Unwrap Frame so that its directions become the x and z axis, than it recomputes the unwrap.
- **Unwrap Frame on ZX**: set the Unwrap Frame so that its directions become the y and z axis, than it recomputes the unwrap.

Keeping control other Unwrap Functions

There are a few things you can do to control your unwrap functions.

- **After you create a new Unwrap Function**: the Tools Set will continue working on that function, until you change your polygons selection. While the function is active, you will be able to change its parameters and even change the function type itself. All the groups for that function will be regenerated with your new function type and parameters.
- **If you select a polygon which was previously unwrapped for a new unwrap function**: it will be removed from the previous Unwrap Function, since each polygon can belong only to one unwrap function at a time. Eventually empty functions and groups will be removed from the part.
- **If you select all polygons and create a new function**: all polygons will be moved to the new function, so all previously generated functions and groups will be removed from the part (you can always Undo things). **This is usefull if you want to remove all the unwraps or restart from scratch**

- **When generating a Function you don't have to select all polygons**: select the ones you want to unwrap now with a specific function, you will be able to apply a second function on the other ones.
- **When you want to load back a function to change some of its parameters**: you can do this by pressing its **Select Function** button in the **Parts Inspector**.

<< Back To Index

UV Panel

With the UV Panel it is possibile to edit and customize UV coordinates. In order to activate it, you need first to look for the **UV Panel** section of the dedicated inspector, and press on **show**.

UV Panel

Curved Poly uses a uv panel which shows unwrapped uv coordinates and which is integrated in the Scene View (rather than being drawn on a separated window). Once you enter the UV Operator the panel will be placed somewhere on the left or on the right of your model, perfectly aligned with your camera. You can then move it, rotate it or scale the way you want using the Unity Transform Tools: in order to do so you may need to point (click) the little orange pivot over the top of the panel, but the pivot is already pointed by default when you first activate the operator (so, you will see directly the transform tools, and the little pivot will be visible only after you point something else)

The panel draws all the edges of the model using its uv coordinates. It also draws different other things depending on the selected mode, which can be:

- **Transforms Mode** : gives you control over unwrap groups transforms. In this case Unwrap Transforms Controls are drawn.
- **Vertices Mode** : gives you control over uv coordinates of vertices. In this case, vertices are drawn as blue squares.
- **Handles Mode** : gives you control over uv coordinates of handles. In this case handles are drawn as magenta squares.
- **Vertices and Handles Mode** : gives you control over uv coordinates of both handles and vertices.
- **Selection Mode** : gives you control over uv coordinates of a group of selected vertices. You can't select vertices in the UV Operator, but you can open a second scene view tab using a different operator to arrange a selection. Or you can switch to Selection Operator, make your selection and get back to UV Operator with a selection active.

The Panel has a few parameters which can be used to customize your experience:

- **Image**: you can assign a texture to the panel background
- **Color**: a background color which is applied to the image if you have one selected.
- **background (UV Panel Settings)**: controls the dimension of the background rendered inside the panel.
- **padding (UV Panel Settings)**: the panel has a gray margin; this parameter controls the size of the padding around the background. Note: it's not a padding amount, its the final size of the most external quad in the panel. The editor will keep it higher than **background**, whatever you do (by, at least, an amount of 0.1).
- **tiling (UV Panel Settings)**: a value applied to scale the texture within the background space. It has no effect if there is not a texture applied.

Editing Unwrap Groups Transforms

Each Unwrap Group has a transform assigned which can be used to edit the position of all uv coordinates belonging to the group at once. Each transform is drawn as a quad on the UV Panel. The quad is a square, by default, the first time an Unwrap Group is generated, and than can change shape. You can edit:

- **The four corners of the Transform Quad**: point (click on the little circle) them to activate them and show the Transforms Tools. Use the **Move Tool** to change the position of the corner.
- **The center of the Transform Quad**: point (click on the little circle) it to activate it and show the Transforms Tools. Use the **Move Tool** to change the position of the center and therefore of all the corners at once. When you activate the center, you can also use the **Rotate Group** and the **Scale Group**, which will affect all the corners too: keep in mind that uv coordinates has only 2 components, and the third component is always set to zero, so any 3D rotation which would rotate points out of the Panel will result in a stretch instead.

In this mode you have access also to two instant actions:

- **Reflect**: Reflect a Transform Group from right to left. You need first to point on one of its 4 corners or on the center. (There is no option to Reflect up to down, but you can reflect right to left and rotate the center by 180°for the same result)
- **Hide By Unwrap Group**: Hide Polygons of hidden Unwrap Groups and Un-Hide Polygons of Unwrap Groups which are not hidden. You need first to access the Geometries Inspector and choose which Unwrap Groups to hide. With this button you will get the groups hidden not only on the UVPanel but also on the model.

(UV) Editing Vertices

In vertices Mode (or vertices and handles mode) you can change the position of each vertex uv coordinate one by one, using the **Move Tool**. Each main vertex is shown as a blue square. You need to point (click) the square to show the tool and start moving the vertex. Elasticities are not applied on the UV-Panel, but the handles will silently follow their vertex (like in Clayish mode).

Also: keep in mind that some vertices may belong to more than one unwrap groups at a time. That applies to all the vertices on the line through which the group is cut during unwrapping. Those vertices will have, consequentially, more than one uv coordinate, which can (have to) be edited one by one.

(UV) Editing Handles

In handles Mode (or vertices and handles mode) you can change the position of each handle uv coordinate one by one, using the **Move Tool**. Each handle is shown as a magenta square. You need to point (click) the square to show the tool and start moving the handle. Any aligned handle is moved as well.

As it happens for vertices, each handle may have more than one uv coordinate, up to two (when the handle belongs to an edge which has been cut during unwrapping processes)

With handles modes you also have access to the following instant actions:

- **Simplify Edge**: you need to point an handle uv-coordinate first. This action will affect both the handles of the edge this handle belongs. This action will convert uv-coordinates for the handles so that the shape of the edge in uv space becomes an exact line. This is generally useful as an initial setup when you plan to manually edit the handles uv coordinates one by one with this tools.

- **Simplify All Uwrap Group Edges:**you need to point an handle uv-coordinate first. This action will affect the uv coordinates of all handles belonging to the same unwrap group of the selected one. This action works like **Simplify Edge** but it simplifies the shape of all the edges in the group. **This is generally useful as an initial setup when you plan to make a custom unwrap by editing the handles uv coordinates one by one with this tools.**
- **Arc Length Edge:** you need to point an handle uv-coordinate first. This action will affect both the handles of the edge this handle belongs. This action will relocate each handle uv-coordinate in order to better redistribute the relative positions of uv coordinates across the edge in the 3D space. This is performed by taking measures of the 3D shape of the edge and use that measures to recompute distances within the edge. This may be a good thing to do on each edge after you have finished moving its handles uv coordinates.
- **Arc Length All Uwrap Group Edges:** you need to point an handle uv-coordinate first. This action will affect the uv coordinates of all handles belonging to the same unwrap group of the selected one. This action will relocate each handle uv-coordinate in order to better redistribute the relative positions of texel across all the edges of the unwrap group in the 3D space. This is performed by taking measures of the 3D shape of each edge and use that measures to recompute distances within each edge. **This may be a good thing to do at the end of any manual editing of uv coordinates of a model.**

(UV) Editing Selections

In Selection Mode you can edit more uv coordinates at once. In order to do so, you need to select a group of vertices with any other tool, then on the UV Panel you can use transform tools to Move, Rotate or Scale all uv coordinates of all selected vertices at once.

> That said: you must pay much attention when you select a vertex with more than one uv-coordinate, because all its uv coordinates will be selected at once (we are going to work on this in future, giving you the option to select which uv-coordinates you want to transform). There are many practical situations in which this make sense, but there are also situations in which this should be avoid. When selecting such vertices does not work, you should rather edit them in vertex mode.

<< Back To Index

Advanced Shaping Tools

> This Tools Sets requires **Curved Poly - Maker**

This **Advanced Shape Tools Set** is an advanced version of **Shape Tools**, meaning that it has all the same tools, plus a set of advanced instruments which will help you working on the shape of your models. There are 8 Advanced tools right now, the 8th having 3 alternatives setup, for a total of 10 buttons:

- **Circleify**: force all selected vertices to lay on a circle.
- **Cylindric Transform**: a transform based on a cylindric volume built over a set of vertices.

- **Curved Tube Transform**: a transform designed to help you model tubular objects, affecting both selected and unselected vertices. It simplifies the way you shape groups of vertices at once.
- **Proportional Transform**: a transform affecting both selected and unselected vertices, unselected vertices transform is weighted based on the distance they have from selected ones.
- **Sparkling Transform**: a transform which automatically replicates shaping actions to do on the handle of a vertex to other vertices.
- **Symmetry**: a transform which defines a set of symmetry relationships between vertices, allowing transforms among them.
- **Mirror**: a transform which Flips selected elements with respect to a mirroring plane.
- **Free Form Transform (FFT)**: a transform which makes you shape an object by moving vertices on a control grid. You have three variants (1x1x1, with 8 total vertices, 2x2x2, with 27 total vertices, and 3x3x3 with 64 total vertices).

All these tools will need some selection to work. Once any operation has been activated, you will enter into an operation state which will block many other actions. In order to get out of the operation and select a new one you can:

- **Apply**: Confirm any change and exit the the operation.
- **Close**: Remove any generated vertex or edge and exit the operation.
- **Press Enter**: like Apply.
- **Press Esc**: like Close.
- **Change Operator (entering another one from the menu on the left of the scene view)**: like Apply.

> So, if you change operator while one of the building operations is active, the operation will be automatically applied.

Circleify

Circleify forces a set of vertices to lay on a circle and than gives you control on the circle parameters. This is very useful if you have a closed path of edges and you want to make it a perfect circle. It is also useful if you have a circle in you model or primitive and you want to shape it without breaking its circular shape. Here the parameters you can change:

- **Position**: the Position of the center of the circle
- **Rotation**: absolute Rotation with respect to a Default circle laying on the XZ plane. This should help you correct the circle orientation when needed.
- **Vertical(%)**: scaling factor applied on the handles of the vertices of the circle. This scaling is applied only on the Vertical direction, that is the direction being at 90 degrees (orthogonal) to the circle plane.
- **rotation**: lateral inclination of the handles of each vertex. This is applied with rotational symmetry (so if two vertices are in opposite direction, this rotation will happen towards opposite directions.)
- **r (radius)**: radius of the circle.

The rotation, Vertical and r can be controlled with magenta arrows in the scene view. Position and rotation can instead be controlled with Unity transform tools. All the parameters can be changed in the menu.

Cylindric Transform

Cylindric Transform is similar to Circleify, but it does not force vertices to take some specific shape. It rather generates a cylindric shaped volumes around selected vertices, and then makes you transform the cylindric volume to reshape all that vertices at once. This is a very useful instrument if you want to create rotational surfaces from cylindric primitives for instance. Here the parameters you can change:

- **Position**: the Position of the center of the cylindric volume.
- **Normal**: The vertical direction of the cylindric volume, which is by default (0,1,0) since this is coherent with the setup of many primitives you can generate with **Create Operator**. This parameter can only be changed in the menu with a manual insertion of values.
- **radius**: radius of the cylinder.
- **hScale(%)**: vertical scale of the cylinder, in the direction of the normal.
- **rotation**: lateral inclination. This is applied with rotational symmetry, and allows you to transform the cylindric volume into a Conic one.

The rotation, hScale and rotation can be controlled with magenta arrows in the scene view. The Position can instead be controlled with Unity transform tools. All the parameters can be changed in the menu.

Curved Tube Transform

Curved Selection Transform allows you to transform groups of points at once in a way that makes transforms on edges between the groups blend smoothly. Before you enter the transform you need to plan carefully which vertices you want to select:

- **Selected vertices will become lead vertices**: lead vertices controls the position of other vertices.
- **Unselected vertices will become passive vertices**: each passive vertex is assigned to one of the lead vertices, depending upon their distances. (There is no other options about this, but we may have more options in the future)

Once active,the operation works in a way which is similar to the pointings operator, but you will be allowed to point only lead vertices. Once you point a lead vertex, all its assigned passive vertices will be selected and also the edges connecting vertices in the group will be selected. The entire group will follow any transform you apply to the lead vertex.

On each lead vertex you can also find 6 transform handles which will allow you to scale or screw the group on its relative x, y or z axis. The handles are 6 because there are separated scaling factor for down and up, left and right, forward and backward.

> This transform always affect all vertices of your model with the exception of the ones which have been hidden, so you may want to hide parts of the model you don't want to be affected by these transforms. That said, it is recommended to use this transform on fresh Curved Polys with only one shape inside: after shaping it and saving the shape, you can always insert it into a more complex Curved Poly with the **Insert** button you can find in **Create Operator**

Proportional Transform

Proportional Transform allows you to transform groups of vertices and make other unselected vertices follow the transform in a smooth, weighted way. Before you enter the transform you need to plan carefully which vertices you want to select:

- **Selected vertices will become full transform vertices**: full transform vertices will follow the transform with a 100% weight. Usually it may make sense to activate the tool with only one selected vertex.
- **Vertices within a specific distance from the selection will become passive vertices**: each passive vertex is assigned a weight which depends on the distance.

- **Vertices at a greater distance will be unaffected**

Once active, it will be possible to transform the selection using Unity Transform Tools, like it happens in the Selection Operator. Full transform vertices will follow the transform, while passive vertices will change their position more slowly, at a pace which depends upon their weight. During the transform every vertex will be drawn has a sphere with a color which blends between gray and magenta: gray means that the vertex has a 0% weight assigned and magenta means that the vertex as a 100% weight assigned. A Distance function is applied to determine such weights with this parameters:

- **Dist (Distance Function)**: the distance at which the weights become 0%.
- **Shape (Distance Function)**: a parameter in the [0,1] range which determines how rapidly the weights will drop to 0%.

You can control both parameters in the menu or through a pair of magenta arrows in the scene view

Sparkling Transform

With Sparkling Transform you can replicate changes you make on a vertex handles to other vertices. You need to select the vertices you want to work on, than the transform will try to match the handles across that vertices in order to apply symmetric changes. There are two relevant cases in which this transform does a good work:

- You select a set of vertices having all the same editing normal and the same amount of handles (this is common if you are working on a Curved Poly primitive).
- You select a set of vertices on a circle. Any change will (hopefully) be applied in a way which is symmetric to the center of the circle.

The transform will work in any case, but it may fail matching handles on different vertices if it doesn't find a good mapping of the handles.

Once active, you can point on a vertex and then you can point on one of its handle to change its length, roll or pitch, the same way you can do in Pointings Operator, but here any change is replicated to all mapped handles on the other vertices, making all selected vertices change in shape at once in the most symmetric way possible (again: this depends on how good is your starting selection so you should have a try and maybe **Close** the operator to roll it back and make a different selection of vertices)

Mirror

Mirroring flips a set of elements mirroring them with respect to a mirroring plane.

How to Use: select any group of polygons. Click on the button to flip it on x direction. Use the controls here to change how your selection is flipped.

Position: Position of the mirroring plane.

Mirror Direction: Mirroring Direction, which is the orthogonal/perpendicular direction the Mirroring plane is facing.

Mirror Direction XY: change the mirror direction to be forward (0,0,1)

Mirror Direction YZ: change the mirror direction to be right (1,0,0)

Mirror Direction XZ: change the mirror direction to be up (0,1,0)

Apply: Close the Mirroring Editor and apply changes.

Close: Close the Mirroring Editor undoing all the changes.

Symmetry

Symmetry defines a set of symmetry relationship between vertices. This relationships are built using a **Mirroring Plane** working as the one used in **Mirroring**. Vertices being ON the plane, are marked with a big yellow spheres. Vertices having a symmetric pair vertex across the mirroring plane are connected to their counterpart with a straight line in the mirroring direction. Moving one of the vertices in the couple, will move the other one as well.

How to Use: select any group of polygons. Click on the button to generate a set of symmetric relationship. If you move a vertex with a symmetric relationship, also the vertex with which it is in relationship will be moved.

Mode: activate/deactivate the editing of the Mirroring Plane. When active, you can change the mirroring plane and vertices relationship are recomputed. When not active, you can't change the mirroring plane, but you can move the vertices around it.

Position: Position of the mirroring plane.

Mirror Direction: Mirroring Direction, which is the orthogonal/perpendicular direction the Mirroring plane is facing.

Mirror Direction XY: change the mirror direction to be forward (0,0,1)

Mirror Direction YZ: change the mirror direction to be right (1,0,0)

Mirror Direction XZ: change the mirror direction to be up (0,1,0)

Apply: Close the Symmetry Editor and apply changes.

Close: Close the Symmetry Editor undoing all the changes.

Free Form Transforms

The Free Form Transform generates a grid of vertices around your selection, so that you can reshape the selection by moving the points in the grid. The grid will be rendered as a set of lines connecting grid points which can be white or blue. You can make white grid points become blue by clicking them. You can then use Unity Transform Tool to translate, rotate or scale the subset of the blue points, while white points will keep unchanged. In the Menu you will find other useful buttons:

- **Reset**: clear the set of blue vertices, making all of them white
- **Extend on Grid (x)**: extend the set of blue vertices on x direction
- **Extend on Grid (y)**: extend the set of blue vertices on y direction
- **Extend on Grid (z)**: extend the set of blue vertices on z direction

Once inside one of the 3 available transforms, you can't change the grid size, but you can always **Close** or **Apply** the transform and open another one.

<< Back To Index

Edit Tools

The **Edit Tools** are by far the most complex Tools Set in Curved Poly Maker. Its main goal is to create custom shapes, that you can subsequently reshape with any shaping tool. Here you can customize shapes, generating new edges and new polygons and changing the relationships between them. For this reason, here you will need to deal with aspects of Curved Polygons which are not directly accessible from other Tools Sets.

> The number of editing tools is planned to grow in future releases. We are planning to release more in subversions 1.3.1 and 1.3.2 which are on their way. The actual tools set is a set of working things, yet it is part of a continuos development to find better ways and solutions for modeling with Curved Poly.

Edit Operator Menu

The Operator has a big menu organized into 5 parts. Each part will be treated in detail in the next subsections, while many tools and

- **(A)** Tools for **Shape Customization**.
- **(B)** Tools which **split or cut elements**.
- **(C)** Tools to **create new Edges and Polygons**.

Furthermore, if you want to use the Editing Tools in depth, you should learn how to **fix broken stuff**, and how to **exploit other tools Sets** in junction with Editing Tools.

Criticalities and Warnings

Some actions and operations available in this Tools Set are pretty much complex and have been tested on many practical situations, yet you may find that your model gets broken if you experiment with them in ways which have not been even considered during development. Here some consideration we want to share with you before you get your models broken:

- You can find some examples of working uses of Edit Operator in this documentation. We are also sharing more through the website and through the youtube channel. Using the Edit Operator the way it is shown on these tutorials should be safe.
- If you want to experiment a way to use the Edit Operator, make a test before breaking a model you have spent hours on.
- Also: while making experiments you should learn to use the **Save New** button often. You may rely on Undo, but when something gets broken its not guaranted the Undo will be able to safely roll back to the previous state of your model.
- If tutorials aren't enough and you have no idea how to get a specific result, you can ask for help at support.mushroomslabs.com. You will find a dedicated thread for this.
- While editing it is much safer to avoid working on a model splitted into more geometries or with unwraps. If you are working on an imported or inserted model, you can use the Materials Tools Set to clean the model from geometries and from unwraps you don't want to deal with when you are here. If you are working on primitives generated with the Create Operator, you should disable the separate and unwraps options (that is by default). If you create a model from scratch in the Edit Operator every new polygon will be generated on the same geometry without using unwraps.
- If you plan to use often the Edit Operator, you should also spend time to understand how you can fix your models. Curved Polys are complex objects, and in many situations it may be that you see bugged features where instead you only have a bad setup of details like Handles, Editing Normals. Most broken models can be fixed by taking control of this details with tools that you can find in the Edit Operator.

<< Back To Index

Shapes Customization

Here the list of actions you can take to customize a Curved Poly Shape:

- **Transform Selection**: this is a simplified version of the **Selection Operator**. Note: this is an operation, (not an operator) and you need to apply (or close) before you can do anything else. You can use it for fast transforms without having to switch to another operator. **Availability: you need to have at least one selected vertex**.
- **Auto Align Handles**: looks for possible alignments across your selected vertices and generate alignment constraints on them.**Availability: you need to have at least one selected vertex**.
- **Quad Align Handles**: force more alignments across any selected vertex, and eventually corrects its handles positions. It works only on vertices with an even number of attached handles and the alignments are generated by matching opposite handles. **Availability: you need to have at least one selected vertex**.
- **Break Alignments**: remove any alignment on selected vertices. **Availability: you need to have at least one selected vertex**.
- **Normalize Handles Directions**: check all smooth vertices in your selection for handles not satisfying the smoothness condition and correct them. Depending on the case, it make make sense to do this after **Fix Normals**. **Availability: you need to have at least one selected vertex**.
- **Fix Normals**: recompute the normals on all selected vertices. This may also change the editing normals on attached edges. **Availability: you need to have at least one selected vertex**.

- **Flip Normals**: flips the normals on all selected vertices and edges. Be careful while flipping normals, normals play a very important role in interpolation algorithms and the shape of polygons attached to that vertices and edges may change in ways you don't want. **Availability: you need to have at least one selected vertex**.
- **Make Vertex Smooth**: Force any selected vertex to become smooth. **Availability: you need to have at least one selected vertex**.
- **Make Vertex Sharp**: Force any selected vertex to become sharp. **Availability: you need to have at least one selected vertex**.
- **Make Edges Linear**: Force any selected edge to become linear. In the process the handles of curved edges will be removed from the model. **Availability: you need to have at least one selected edge**.
- **Make Edges Curved**: Force any selected edge to become curved. In the process two new handles will be generated for any linear edge. If the vertices of the edge are smooth, the new handles will be placed in order to satisfy the smoothness condition for handles. **Availability: you need to have at least one selected edge**.

<< Back To Index

Shapes Customization

Tools for Removing Elements

- **Remove Doubles (and Weld Edges)**: activate the Removed Doubles operator which can also weld Edges. **This is one of the most important tools in the Edit Tools**. **Availability: you need to have something selected.**
- **Cancel**: cancel any selected element. **Availability: you need to have something selected.**
- **Cancel Vertices**: cancel any selected vertex. **Availability: you need to have some selected vertex.**
- **Cancel Edges**: cancel any selected edge. **Availability: you need to have some selected edge.**
- **Cancel Polygons**: cancel any selected polygon. **Availability: you need to have some selected polygon.**

Tools for Cutting Elements

- **Circular Hole Operation**: cut an hole around a vertex. The hole can be subsequently used as a starting point for other tools. Despite the name, the cut is not necessarily a circle and it may be useful to use **Circleify**. **Availability: you need to have 1 (and only 1) vertex selected.**
- **Subdivide Edge Operation**: split an edge into two edges. This split can be extended through adjacent polygons, splitting both edges and polygons. **This is one of the most important tools in the Edit Tools**. **Availability: you need to 1 (and only 1) edge selected.**

<< Back To Index

Making Shapes

Warning! This section contains some informations and/or images which were produced in version 1.2 docs. Therefore, this page shall be considered as a draft, the intended version 1.3 page will be ready soon.

This is where to start if you want to make a model from scratch.You will find more informations about this in the **section dedicated to Custom Primitives**

- **Put Edges**: activate the Put Edges Operation which allows you to draw Curved Edges.**This is one of the most important tools in Edit Tools**.
- **Put Lines**: activate the Put Lines Operation which allows you to draw Linear Edges.
- **Put Vertex**: activate the Put Vertex Operation which allows you to add separated vertices.
- **Make a Circle**: activate the Make a Circle Operation which allows you generate a circle connecting two selected vertices. **Availability: you need to have 2 (and only 2) selected vertices**
- **Make Polygons**: automatically generate polygons. It only generates polygons attached to any of your selected vertices. Possible candidates for polygons generation are found by Curved Poly starting from the Corners on the selected vertices: if somethings goes wrong, you may have some **corners or bindings to fix** . **This is one of the most important tools in Edit Tools**. **Availability: you need to have at least one selected vertex**
- **Duplicate**: activate the Duplicate Operation which allows you duplicate your selection. The duplicated

elements can then be moved around before you confirm the operation. **This is one of the most important tools in Edit Tools.Availability: you need to have at least one selected vertex**.

- **Flip Polygons**: flip all selected polygons. **Availability: you need to have at least one selected polygon**.
- **Create Hole Joints**: create a set of polygons which fills an hole. The hole should be setup to be made of a closed set of aligned handles, otherwise the hole will only be filled partially. You only need to select one of the edges in the hole to activate this, but you may use **Grow Through Alignments** to check if the hole is made of a closed set of aligned handles first. **Availability: you need to have at least one selected edge**
- **Create Bridge Joints**: create a set of polygons which connects two separated holes. The holes should be setup to be made of a closed set of aligned handles, otherwise the holes will only be connected partially. You only have to select one of the edges of each the two holes to activate this. **Availability: you need to have exactly 2 (and only two) selected edges.**
- **Extrude Path**: create a set of polygons which extrude a path of aligned edges. Any selected edge will be automatically extended through its alignments. Only edges having no more than one polygons attached can be extruded. **Availability: you need to have at least one selected edge.**

Making Custom Primitives

Apart from the shapes you can generate with the **Composition Tools**, you can use the **Edit Tools** to generate new primitives or parts for you models in many ways. There are many approaches you can follow here:

- **(A)Generate Edges and Make Polygons**: you can create a model from scratch by modeling Edges (or Lines or Vertices) in the scene view. With two selected vertices you can also generate Circles. Once a you have a good set of edges and vertices you can fill it with polygons with the **Make Polygons** button which will guess the best polygons to fill your net of edges. **Make Polygons** take into accounts all the constraints required to make interpolation processes work well, so that the final mesh extracted from the polygons looks good.
- **(B)Modify a Shape with Cuts and Extensions**: you can start from a primitive or model you already have and delete parts you don't need with Canceling Buttons. With one vertex selected, you can also create a hole; when you have an hole (either generated after removing contents or with the Hole Operation) you can exploit the Extrude Path or the Hole Joints operations to add polygons to your shape.
- **(C)Add Control Vertices**: the best tool to add vertices (and edges and polygons) to a shape you already have is the **Subdivide Edge Operation**. This operation works on one edge at a time and allows you to split it into two halves. You may then extend it to adjacent polygons and to more edges in the case the polygons are quads. If you need something more specific which can't be accomplished with that operation, you can also rework parts of you shape by removing polygons or edges and using Put Edges + Make Polygons to generate different parts and details.
- **(D)Combining and Stitching Shapes together**: you can take two (or more) shapes of your choice and put them together in the same model. You can do this by using the **Composition Tools**, or you can also duplicate parts in the **Edit Tools**. Then, if they have an open part (eventually one generated with cutting tools), you can try to stitch/weld them together. You have more alternative ways to do so. One approach is to use **Remove Doubles** which also weld edges when their shape is approximative the same. Another solution is to take select two open also facing each other and go for

Bridge Joints. You can also use Put Edges and Make Polygons to manually build a custom Joint between two holes in two separated surfaces.
- **(E)Import a LOW POLY Mesh and Use Fixing Tools to Transform it into a Curved Poly**: be sure to have the time to understand all the passages and the use of fixing tools. The mesh must be have a very low amount of faces, since each face will be transformed into a Curved Poly Polygon.

Here we will show a few examples of all this approaches. You can also look for Curved Poly tutorials online to learn more. Here an index of the examples:

- **(A.1)Put Edges and Make Polygons**
- **(A.2)Manually create curved tubular surfaces with Make a Circle**
- **(A.3)Put Lines and Make Polygons**
- **(B.1)Cutting Parts of a Shape**
- **(B.2)Hole Operation**
- **(B.3)Extrude Path and Hole Joints**
- **(C.1)Subdivide Edge Operation**
- **(C.2)Manually Subdivide Edges or Polygons**
- **(D.1)Duplicate a Part and Weld the two clones together with Remove Doubles**
- **(D.2)Connecting two separated Parts with a Bridge Joint**
- **(D.3)Manually Connect two Parts with Put Edges and Make Polygons**
- **(E)Import LOW POLY Meshes to transform into Curved Polys**

(A.1)Put Edges and Make Polygons

In this section we will discover more about **Put Edges** and **Make Polygons**

With put Edges you can add vertices and Edges directly on the scene view. Before you start you should watch you **Pivot**

Position: you will find the Pivot in the Scene View rendered as a yellow spheres surrounded by two circles, representing something which should look like a target. You can click on the target and use the **Moving Tool** to position it. When you add vertices with **Put Edges** a vertex position will be computed taking into account the mouse position and the depth of the pivot from the scene view camera perspective. So, for instance, if you place the pivot in (0,0,-1.5) and you choose to align the scene view Camera to X-Y axis (that is with the back and front view), every vertex generated with Put Edges will have a z coordinate equal to -1.5, and x and y coordinates computed from mouse position.

Entering and Exiting Put Edges Operation

Once you click on **Put Edges** you will enter in an operation state from which you can exit in either one of these ways:

- ✓ **Apply**: Confirm any change and exit the the operation.
- ✕ **Close**: Remove any generated vertex and exit the operation.
- **Press Enter**: like Apply.
- **Press Esc**: like Close.
- **Change Operation (entering another one from the menu on the left of the scene view)**: like Apply.

Generating a Net of Edges

Once **Put Edges** has been activated you will be able to add vertices by clicking on the scene view. An edge will be generated between two consecutive edges, generating a continuous piecewise curve. You can break the sequence and start a new one by pressing **Space**, or you can also **Apply** and re-enter pressing **Put Edges** again.

If you position your mouse on any existent vertex , its look will change: in this case **Put Edges** will re-use that vertex instead of generating a new one: you can exploit this to connect each vertex with more than two edges, so that you can create a complete net of edges. When a vertex is smooth and it has already more edges attached, new edges may be reshaped to conform (automatically) smoothness conditions for that vertex.

Making Polygons

Once you have a net of Edges you can start generating polygons. To do so you first need to select at least a vertex: Curved Poly will generate polygons by looking for any closed loop of edges nearby your selected vertices. Then: you only have to press **Make Polygons** and everything is automatic. A faster approach is to press **Select All**, and then **Make Polygons**, in this way any possible polygon available in the model will be generated. You can always cut unrequired polygons at a second time with **Cancel Polygons**. You may also **Flip Polygons** if they are facing the wrong side.

> You may need to practice a bit with this before being able to use correctly **Make Polygons**. There are situations in which polygons can't be generated because there are edges with a bad shape, or with overlapping handles, which are blocked by Curved Poly to avoid issues with interpolation algorithms. When you think you did good but your polygons couldn't be generated, you can try go through an inspecting and fixing process. This is explained more in deep in a **Dedicated Chapter**

(A.2)Manually create curved tubular surfaces with Make a Circle

In this section you will discover how to use and exploit **Make a Circle** while working with **Put Edges** and **Make Polygons**

Setup

Let's start drawing two set of edges with **Put Edges**. Then we get out from put edges and we select two vertices, one for each vertex.

Make a Circle

With Two selected Vertices you can enter **Make a Circle**. By pressing the button you will enter into an operation state. Here you can:

- **Choose** the amount of edges used to make the circle in the Panel named Circles Sizes.
- ✓ **Apply**: Confirm any change and exit the the operation.
- ✕ **Close**: Remove any generated vertex or edge and exit the operation.
- **Press Enter**: like Apply.
- **Press Esc**: like Close.
- **Change Operation (entering another one from the menu on the left of the scene view)**: like Apply.

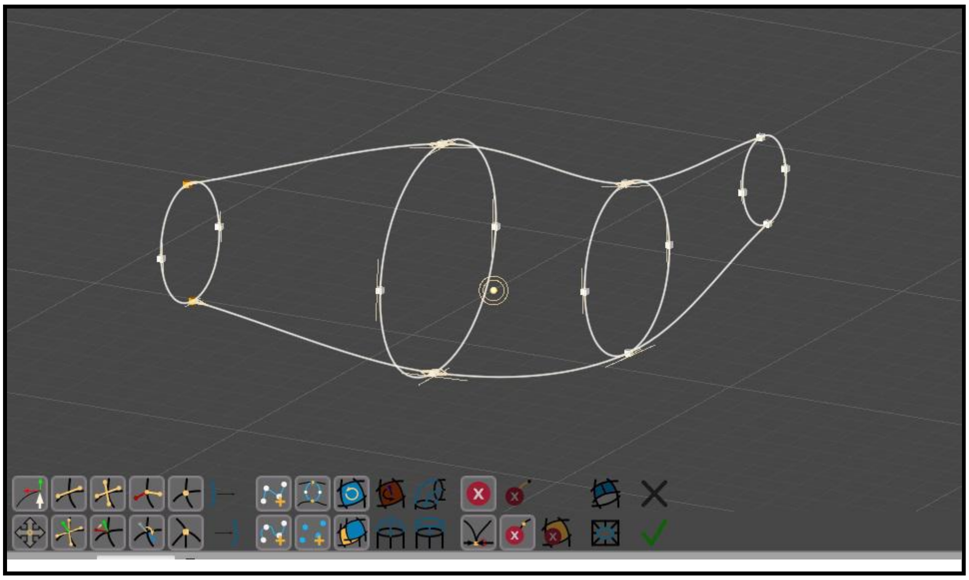


Fig. 110 Making more Circles

You can then use the operation more and more times, but you will need to exit and enter each time if you want to generate more circles.

Put more Edges and Make Polygons

You can add more edges to connect the addictional vertices generated with **Make a Circle**.

Now we **Select All** and make polygons. If there aren't issues with the edges handles, you should see a custom tubular surface following the two curves you have drawn from the beginning. Know that you may need to use **Flip Polygons**.

(A.3)Put Lines and Make Polygons

Put Lines works exactly like **Put Edges**, with a few important differences:

- **Lines are generated**: instead of a continuous smooth curves, you will be able to generate a polyline with no alignments. Press **Space** to break the chain of vertices and generate a new polyline.
- **Generated Vertices are Sharp** by default instead of being smooth.
- **If you reuse a smooth vertex it will become sharp**.

Here each vertex is forced to be sharp, but you can change this at a second time.

> **Put Edges** and **Put Lines** both work in ideal situations where everything is smooth if you Put Edges and everything is sharp if you Put Lines. That said, you can mix things up, making models which have linear parts and smooth parts. You can also have Curved Edges with Sharp Vertices and Linear Edges with Smooth Vertices. Mixing things this way is possible, but it requires a

> greater comprehension of a few mechanisms which will be discussed in the **Inspecting and Fixing**.

(B.1)Cutting Parts of a Shape

Cutting parts of a Shape is pretty straightforward. You need first to select the parts you want to remove, then you can:

- **Remove Doubles (and Weld Edges)**: merge elements which are closer than a specific distance that you can control. **Availability: you need to have something selected.**
- **Cancel**: cancel any selected element.
- **Cancel Vertices**: cancel any selected vertex.
- **Cancel Edges**: cancel any selected edge.
- **Cancel Polygons**: cancel any selected polygon.

About Remove Doubles

The **Remove Doubles** will put you in an operation state like similar operations we have seen around, and you will need either to **Close** or **Apply** it to exit.

While you use **Remove Doubles** you can setup the distance **d** more and more times to fix your model. Every time you do this, you need also to press on the **Update** button to make **Remove Doubles** recompute the cut with the new distance. The total amount of removed Vertices and Edges will be shown in the same panel below the distance **d**.

(B.2)Hole Operation

> NOTE: as Version 1.0 Hole Operation still needs some works. It works most of the times, but often generated polygons need some fixes afterwards. Since issues

happen mainly on polygons, a general solution is to call **Cancel Polygons** with a selection of polygons looking bad and call **Make Polygons** soon afterwards with the same selection. Or, since often the problem is that polygons are flipped, you can try to correct them with a **Flip Polygons**. Another important tip is to avoid pre-unwrapped models on this at all (unwrapping brings an additional degree of complexity in parts and you should avoid complex editing operation on parts splitted into Unwrap Groups).

(B.3)Extrude Path and Hole Joints

When you have an open surface you can use **Extrude Paths** or **Hole Joints** to add more polygons. This two actions (together with **Bridge Joints** which is discussed in point **(D.2)**) are a preview of a **paths** systems which we are going to introduce in one of the next versions of Curved Poly Maker. At the moment, they do their job well only if you make a good setup first: technically speaking, you need to have a closed set of edges, each one with a polygon on one side and free on the other side, and it may be better if you have put alignments first on that edges through the border. Basically, it has to be a closed border for an open surface. All the edges must be curves (lines may work, but it's not guaranteed), all the vertices must be smooth (sharp vertices may work, but it's not guaranteed). For example, this is the case if you start with a Cylinder Border primitive, or if you have generated a tubular surface exploiting **Make a Circle** as explained in **(A.2)**. If this is your case, you need to select one of the edges of the border before activating either Extrude Path or Hole Joints (but you can select all the edges on the border if you like, or even some of them): since the edges are aligned, the software will automatically grow the selection through the border. Know that this two operations may still try to do something in any case once you have at least an edge selected: if you don't like the result you can always **Undo** it.

Extrude Path

Extrude Path add a set of new polygons, one for each edge, which extrudes the open border.

Hole Joints

Hole Joints try to close the hole with a cap. To do so, it will try to guess a good position for a new vertex which will be added to the model, and will connect that vertex with the border by adding a new set of edges and polygons.

(C.1)Subdivide Edge Operation

Subdivide Edge allows you to split edges and polygons to add more details, increasing the amount of control vertices.

Polygons Sides With more Edges

The side of a polygon can have more than one edge. Furthermore, a Group of edges can be used as one unique side for a polygon, and as separated sides for other polygons, making the polygons form a **T-Junction**. While planning your model, keep in mind that (apart from technical issues derived by the use of each operation) the only constraint here is that each edge can belong to a maximum of two sides of two different polygons, one on its **left** and one on its **right**, where **left** and **right** are computed based on the direction of the editing normals of the two extreme vertices of the edge.

Subdivide Edge Operation

The **Subdivide Edge Operation** will put you in an operation state like similar operations we have seen around, and you will need either to **Close** or **Apply** it to exit.

You need to select one (and only one) edge before entering the operation. That edge will be automatically split into two halves. After that you will be able to:

- **Control the cut position**: by either dragging the arrows on the scene view or changing the **t** parameter in the **Position** panel in the menu. Changing this parameter will not directly change the cut on the model, you will have to do an **Update** once you have assigned a position you like.
- **Extend through Polygons**: This allows you to extend the cut through polygons adjacent to the edge, by cutting both the polygon and the successive edge in a precomputed sequence. This sequence is evaluated at the beginning, when you press the **Subdivide Edge Operation** button, and it can only contains a successive set of adjacent curved quadrilateral polygons; it can also contain up to 2 triangular polygons, since the cut through the polygon will fall into an opposite vertex, instead an opposite edge, ending the polygons sequence (it's complex to explain through words, it becomes more intuitive if you check the figures and try it in the editor).

You can cut a model as many times as you want. An important limit to keep in mind is that the tool actually cut polygons sides, not edges. So: if a polygon side has only one edge, everything will go as explained. If you try to cut a single edge being part of a side of a polygons using more edges, the software will still cut the side and not the edge (so you can't cut further an edge with this tool if you do not extend the cut through polygons each time). If you want a polygon to have a side with more than two edges, you need to do that manually following the steps in **(C.2)**

(C.2)Manually Subdivide Edges or Polygons

You can of course subdivide edges and polygons manually. This will take some time, but not too much when you know what to do. A subdivision process basically should follow this two steps:

- **Delete parts you want to subdivide**. You should do this using planning your selections carefully. This is also a good time to use **Cancel Polygons** or **Cancel Edges** rather than the most general **Cancel Edges**.
- **Regenerate the Edges you wanted to subdivide with Put Edges**. Once you learn to use **Put Edges**, it's pretty fast to regenerate removed edges. Here you can rebuild edges you canceled using a greater number of edges for instance.
- **Make Polygons**. After you have finished rebuilding the structure of your model, rebuild everything by selecting the new Edges (actually you need to select vertices, not edges: you can do this with **Extend To Vertices** if you think you miss something) and press **Make Polygons**.
- **Reshape**. In the end, you will need to correct the shape. The editing normals on vertices and edges will be most likely bad; you can try to use **Fix Normals** to correct them. You may also experiment different edge normals setups using the **Advanced Shape Tools**. Shape Control may also help to correct circular or rotational surfaces. For everything else, you will step back to **Shape Tools** to fix the elements.

(D.1)Duplicate a Part and Weld the two clones together with Remove Doubles

Here we will see how to use **Duplicate** and **Remove Doubles** to increase the complexity of a model

About Duplicate

Duplicate Operation allows you to duplicated a part of your model. You need to select something first (even a single vertex, or the entire shape). Pressing the button will put you in an operation state like similar operations we have seen around, and you will need either to **Close** or **Apply** it to exit.

- After you press the button, **a new clone of your selection will be generated and will become selected in place of the original selection.**
- Before getting out of the operation, **you can use Unity Transform Tools to move, rotate or scale the cloned elements**.
- **You are free to change your selection if you need**, the tool will always affect the originally cloned elements.

> **Reflected Duplicates**: there is no direct way (as version 1.0 of the Maker) to Reflect a Duplicate. The most direct sequence is : in **Duplicate Operation** (or also in **Transform Operation**) manually write **-1** in the of field of a scaling component, either x, y or z depending on the reflection result you want to accomplish. Then exit the **Duplicate Operation**(or the **Transform Operation**). With the same polygons selected (correct the selection if necessary), press on **Flip Polygons**. Finally: press on **Fix Normals** to correct issues with normals.

Weld the Duplicates

If you are duplicating a shape with an open part, you can than exploit **Remove Doubles** to Weld them together.

(D.2)Connecting two separated Parts with a Bridge Joint

When you have two open surface you can use **Bridge Joints** to connect them. This action (together with **Hole Joints** and **Extrude Path** which are discussed in point **(B.3)**) is a preview of a **paths** systems which we are going to introduce in one of the next versions of Curved Poly Maker. At the moment, it does its job well only if you make a good setup first: technically speaking, you need to have two closed set of edges and the edges must have all one polygon on a side and nothing on the other side; it may be better if you have put alignments first on that edges through the border. Basically, they have to be two closed borders for two open surfaces. All the edges should be curves (lines may work, but it's not guaranteed), all the vertices should be smooth (sharp vertices may work, but it's not guaranteed). For example, this is the case if you start with a Cylinder Border primitive, or if you have generated a tubular surface exploiting **Make a Circle** as explained in **(A.2)**. If this is your case, you need to select one of the edges on each border before activating Bridge Joints (they must be only two edges, or the button will be disabled): since the edges are aligned, the software will automatically grow the selection through the border. Know that this operation may still try to do something in any case once you have at least two edges selected: if you don't like the result you can always **Undo** it.

(D.3)Manually Connect two Parts with Put Edges and Make Polygons

You can always connect two open parts with **Put Edges** and **Make Polygons**.

(E)Import LOW POLY Meshes to transform into Curved Polys

You can use **Composition Tools** to insert a Mesh in your model. This is performed with the **Insert Mesh** button. You will have to choose a mesh in your project. After the import a curved poly will be generated:

- **All Made of Lines**, so no Curved Edges
- **All Made of Sharp Vertices**, so without any kind of smoothness
- **All Made of Triangles**, directly imported by the triangles array in the mesh.

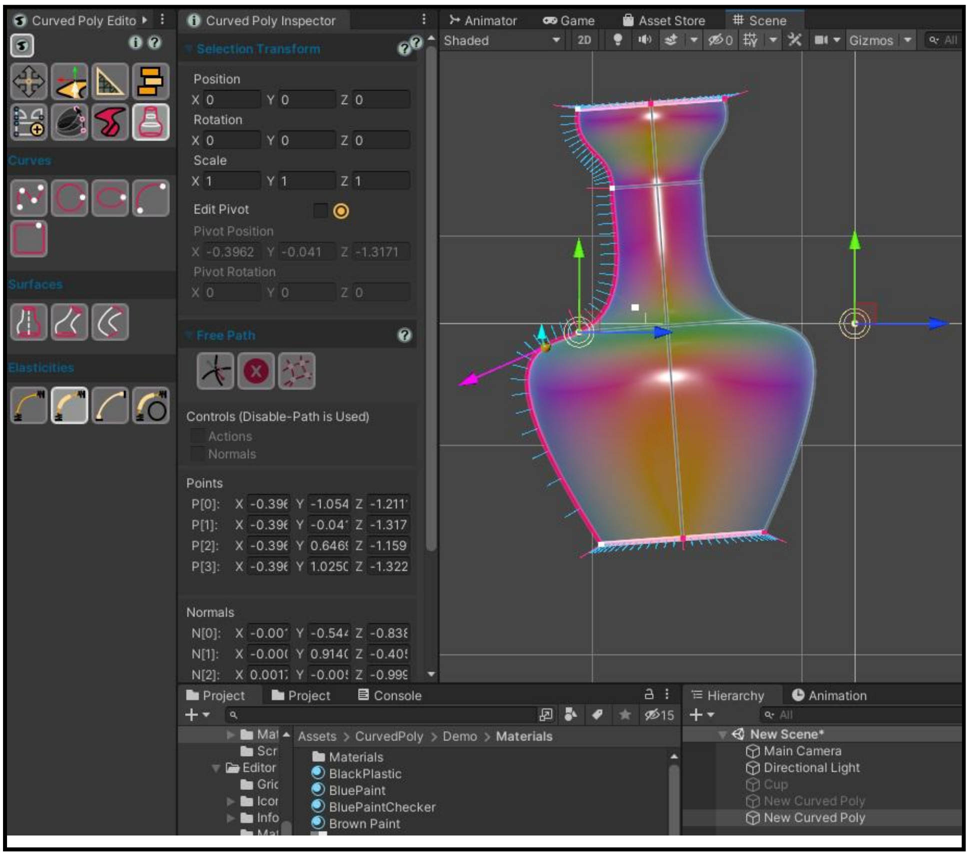
So, after the import, you should consider using:

- **Cancel Edges** to remove edge between two adjacent triangles, so that you can create a quad soon afterwards with **Make Polygons**.
- **Make Edges Curved** to make all the linear edges curved
- **Make Vertex Smooth** to make all the vertices smooth
- **Fix Normals**: to force a recomputation of editing normals on edges and vertices; this may be necessary after using **Make Edges Curved** or **Make Vertex Smooth**.
- **Flip Polygons**: to flip polygons which look watching the wrong side.
- **Normalize Handles Direction**: to correct handles on smooth vertices which you think are not set up correctly.
- **Auto Align Handles**: to generate alignments through all the model (there are many tools which are based on alignments, so this is always a good idea).

The process of converting a mesh into a Curved Poly may become a complex stuff in some situations. Refer to the chapter **About fixing stuff** for some more deep instructions.

<< Back To Index

Curves and Surfaces



Position
X 0 Y 0 Z 0
Rotation
X 0 Y 0 Z 0
Scale
X 1 Y 1 Z 1
Edit Pivot
Pivot Position
X -0.3962 Y -0.041 Z -1.3171
Pivot Rotation
X 0 Y 0 Z 0

Controls (Disable-Path is Used)
Actions
Normals

Points
P[0]: X -0.396 Y -1.054 Z -1.211
P[1]: X -0.396 Y -0.04 Z -1.317
P[2]: X -0.396 Y 0.646 Z -1.159
P[3]: X -0.396 Y 1.0250 Z -1.322

Normals
N[0]: X -0.00 Y -0.544 Z -0.838
N[1]: X -0.000 Y 0.9140 Z -0.405
N[2]: X 0.0017 Y -0.00 Z -0.999

Fig. 132 Curves and Surfaces Tools Set

With Curves and Surfaces you will be able to generate advanced primitives and shapes for your Curved Polys. The usual workflow for Curves and Surfaces works as follow:

- **Make one or more curves** : curves are used to shape surfaces, and you need to make at least one curve to generate a surface
- **Make a Surface** : surfaces are generated from curves and other parameters assigned to them.
- **Convert into a Curved Poly** : a Surface can be converted into a set Curved Poly Edges and Polygons, making it accessible to other Tools Sets.

Surfaces and Curves works on a different layer then the rest of Curved Poly elements. You cannot edit edges or polygons when you are using Curves and Surfaces Tools and you cannot edit Surfaces or Curves when you are using the other Tools Sets. **Convert into Curved Poly** transform the surfaces into Curved Poly polygons making them available in the other tools sets - and no more available in Curves and Surfaces Tools Set.

Important! Surfaces and Curves are supposed to be saved with the Editable Asset of your curved poly, but actually they are stll not. This is planned to be introduced in one of the upcoming versione 1.3.x, prior to the release of version 1.4

Curves

There are more Curves you can use. Apart from the Free Path, all the other ones are made with different settings of the same tool, so you can start with a circle and transform it into a Round Rect for instance.

Free Path: free paths are splines, similar to the ones generated with Put Edges in Edit Tools.

Circle: creates one or more circles. Circles are shaped paths: after creation they can be transformed with parameters shared with arcs, ellipses and rects.

Ellipse: creates one or more ellipses. Ellipses are shaped paths: after creation they can be transformed with parameters shared with arcs, ellipses and rects.

Arc: creates one or more arcs. Arcs are shaped paths: after creation they can be transformed with parameters shared with arcs, ellipses and rects.

Round Rect: creates one or more round rects. Round Rects are shaped paths: after creation they can be transformed with parameters shared with arcs, ellipses and rects.

Surfaces

Surfaces are generated from on or two Curves. Surfaces using two curves need the two curves to match somehow: they must have the same number of sides and be either both closed curves or open curves.

Revolution Surface: creates a revolution surface, making any path to rotate around an (editable) axis. After pressing the button, it is necessary to choose a path in the scene view.

Curved Tube Surface: creates a tubular surface connecting two paths. You must have two paths with the same number of edges first.

Loft Surface: creates a loft surface connecting two paths. You must have two paths with the same number of edges first.

<< Back To Index

Free Paths

Free paths are curved splines used to generate surfaces. When you generate a Free Path you only need to click in Scene.

CLICK to add new vertices to the path

DOUBLE CLICK to close a path and generate a new one

Press ENTER or ESC to exit generation mode

After a Free Path is generated, you can click on it for further editing. Curves and Handles of the path works pretty much the same way as Curves and Handles you can edit in Shape Tools, but here you have further options.

Flip Normals: flips the editing normals of the path.

Cancel Curve: cancel this curve.

Convert into Curved Poly: transform this curve into a regular Curved Poly set of edges, which can edited from the other tools.

Free Paths Controls: Actions

Controls: Actions Activate this to access a set of useful in-scene buttons to edit the path:

+: this buttons add a vertex in the middle of an edge

-: this buttons remove a vertex and merge the edges around it.

C: this buttons make an open path to become close by connecting its first and last vertex.

Free Paths Controls: Normals

Activate this to access a set of in-scene tools to edit the path normals.

Activate this to access a set of in-scene tools to edit the path normals.

Free Path Data

Points : list of points the path is made of.

Normals : list of the normals of each point.

Edge Normals : list of the normals of each edge in the path.

<< Back To Index

Parametric Curves

Most of the Curves are merged into a unique parametric model, which can be used to blend a circle into an ellipse into an arc or into a rounded rect, or a mix of all of them. You shall try the parameters to discover what you can create here. When you create one of this curves, it will be shaped into one of this possible presets:

- **Circle**: A circle is obtained when most of the parameters are kept at their default values.
- : Ellipses add some stretch to the default Circle.
- : Round Rects are generated by adding some more stretch to the vertices between each segment of the Curved. Increasing the number of sided allows to change the Round Rect for instance into a Round Pentagon or Hexagon.
- : An arc is a subset of a Circle constrained by two angles.

On each curve, you can take one of this actions:

- **Flip Normals**: flips the editing normals of the path.
- **Cancel Curve**: cancel this curve.
- **Convert into Curved Poly**: transform this curve into a regular Curved Poly set of edges, which can edited from the other tools.

Parameters

N: number of edges (sides) used on the Curve

Center: use this to edit the center of the curve.

Orientation: rotate the curve.

Bounds-size: overall size of the curve. For instance, it controls the radius of a circle or the axis of an ellipse.

Bounds-stretch: stretch the shape in one direction. For instance this can be used to transform a circle in an ellipse.

Angles-0: first angle on which an arc is defined. This can be used, for instance, to transform a circle into an arc. Default value is 0.

Angles-1: second angle on which an arc is defined. This can be used, for instance, to transform a circle into an arc. Default value is 6.283185 (sorry for radians!).

Raps-hRap: first value used to blend a circle into a Round Rect. Try it!

Raps-wRap: second value used to blend a circle into a Round Rect. Try it!

Raps-starRap: add an oscillation to the shape of the curve, allows the generation of star-shaped curves (use N to define the amount of tips of the star)

<< Back To Index

Creating new Surfaces

Revolution Surface

Create a Revolution Surface, that is a surface generated by rotating a Path around an axis. Both the path and the axis can be edited at a second time.

How to Use: Activate the generator by pressing the button. Choose a Path by clicking on its **Path** icon in scene. Press again in scene nearby the path to place the revolution axis. You will be able to correct the result at a second time.

Loft Surface

Create a Loft Surface, that is a surface generated by connecting two paths with the same amount of edges with raw lines.

How to Use: Activate the generator by pressing the button. Choose a Path by clicking on its **L(Left)** or **R(Left)** button in scene. Choose a second path by pressing again on the **L** or **R** button: the second time only paths having the same amount of edges as the first path are available. In case there are no such paths, you can abort with ESC.

Curved Tube Surface

Create a Curved Tube, that is a surface generated by a flux of circles connecting two paths with the same amount of edges. Circles are larger or smaller according to how much the paths get closed.

How to Use: Activate the generator by pressing the button. Choose a Path by clicking on its **+** button in scene. Choose a second path by pressing again on the **+** button: the second time only paths having the same amount of edges as the first path are available. In case there are no such paths, you can abort with ESC.

<< Back To Index

Updating Surfaces

Important!All surfaces will update when their curves are updated. To change the shape of the curve a surface is made of, you need to click the curve, not the surface.

Warning!: once a curve has been assigned to a surface, it is still not possible to properly change its number of sides. Also, it is not possible to change the number of sides of a surface. We are going to add this features in one of the upcoming versions 1.3.x, before the release of version 1.4.

Revolution Surface

Click on the path the Revolution Surface is made of to edit it.

The Rotation Axis appears in scene as a green line with an upper green handle and a lower blue handle. There is also a Transform Tool in the middle. You can edit and interact with the axis by using any of the handles in scene.

Editing Revolution Surface

Click on the surface of the Revolution Surface to interact with it.

Flip Surface: flips the side of the surface, by flipping its normals and changing the side its polygons are watching

Cancel Surface: cancel this surface, freeing the paths its made of

Convert into Curved Poly: transform this surface into a regular Curved Poly Shape, which can edited from the other tools.

Center: use this to edit the center of the curved tube

Rotate: rotate the surface (and its paths) around the Center.

Loft Surface

Click on one of the paths the Loft Surface is made of to edit them.

Click on the surface of the Loft to interact with it.

Flip Surface: flips the side of the surface, by flipping its normals and changing the side its polygons are watching

Cancel Surface: cancel this surface, freeing the paths its made of

Convert into Curved Poly: transform this surface into a regular Curved Poly Shape, which can edited from the other tools.

Center: use this to edit the center of the loft surface.

Rotate: rotate the surface (and its paths) around the Center.

Scale: scale the surface (and its paths) with respect to the Center.

Curved Tube Surface

Click on one of the paths the Curved Tube is made of to edit them.

Click on the surface of the Curved Tube to interact with it.

Flip Surface: flips the side of the surface, by flipping its normals and changing the side its polygons are watching

Cancel Surface: cancel this surface, freeing the paths its made of

Convert into Curved Poly: transform this surface into a regular Curved Poly Shape, which can edited from the other tools.

Center: use this to edit the center of the curved tube

Rotate: rotate the surface (and its paths) around the Center.

Scale: scale the surface (and its paths) with respect to the Center.

<< Back To Index