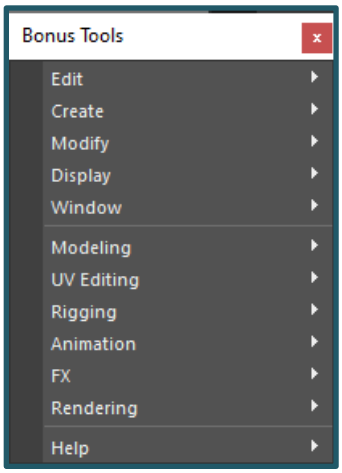
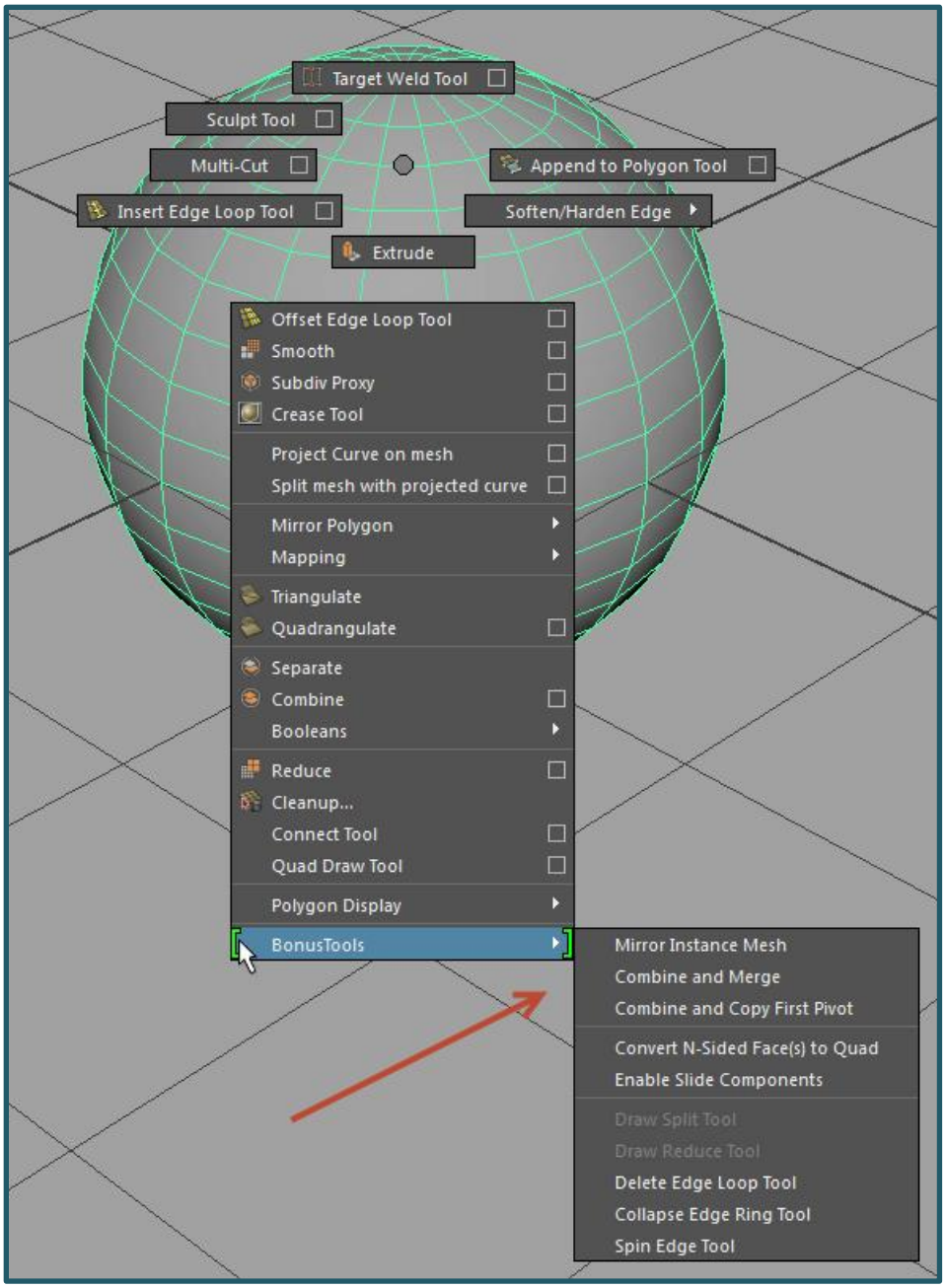
AUTODESK® MAYA®
BONUSTOOLS 2025

Overview

Bonus Tools 2025 is a free collection of useful Maya scripts and plug-ins. After installing Bonus Tools, an additional pull-down menu will be added to the end of the main Maya menu. This new menu provides easy access to a variety of tools and utilities for daily use. Maya Bonus Tools has been organized to mimic the layout of the standard Maya menu sets. Each sub-menu contains a number of related tools. Each of these can be torn off and floated just like standard Maya menus.



Additionally, various modeling related BonusTools are integrated into the standard Shfit+RMB marking menus. This gives power modelers fast, context sensitive access to all modeling functionality contained within BonusTools, without having to break out of your modeling workflow. Simply select a mesh, face, edge or vertex and hold Shift+RMB.

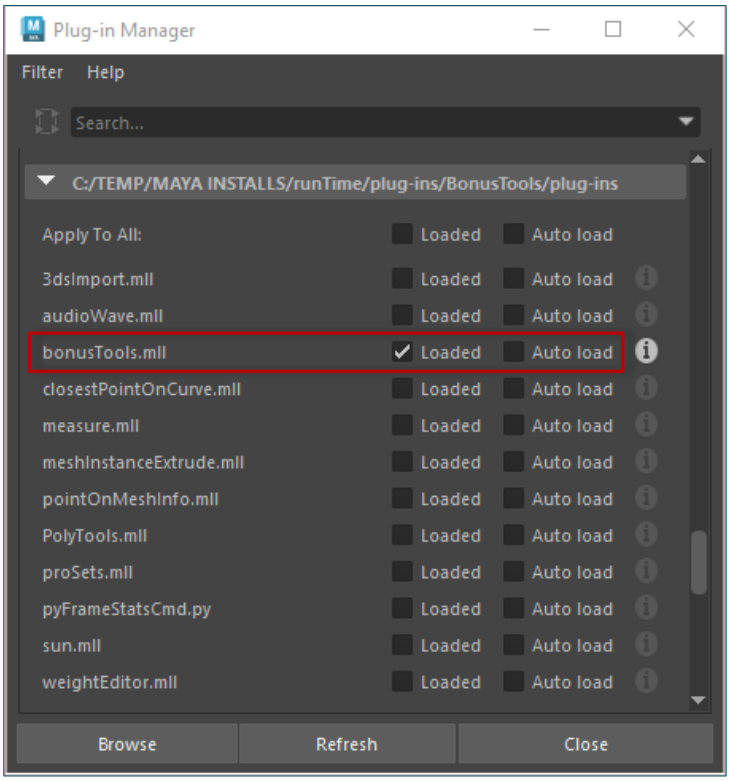


Bonus Tools are provided to you on an "as-is" basis. Autodesk® does not provide any direct support in connection with this collection. However, submission of problems and suggestions are encouraged and can be logged through from the Maya Help menu with Report a Problem or Suggest a Feature.

Notes on BonusTools 2025

Improvements

- BonusTools now automatically installs as part of the regular Maya 2025 installation.
- Simply load the BonusTools plugin from the Plugin Manager and agree to the usage terms.



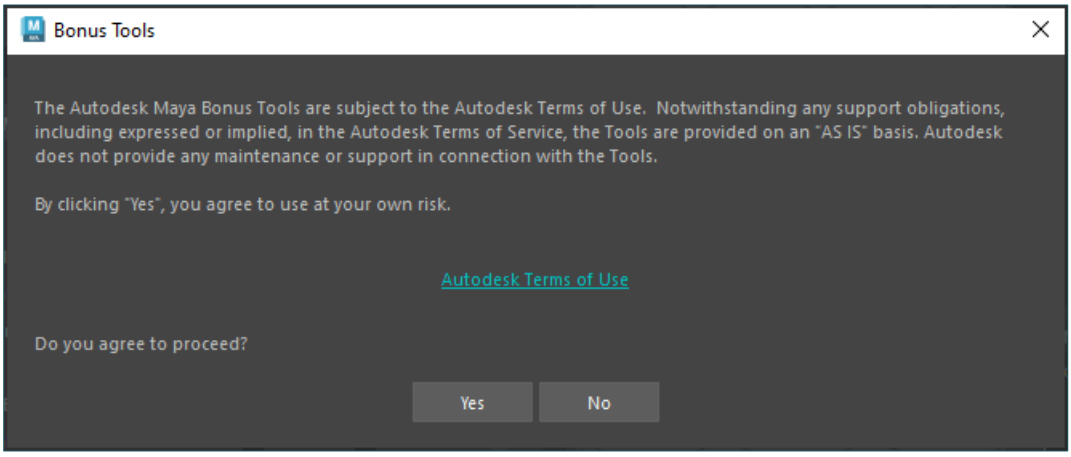
Removed Tools

Display – Enable Popup Help (the bug that this was intended to help with has been fixed in Maya)

Contents

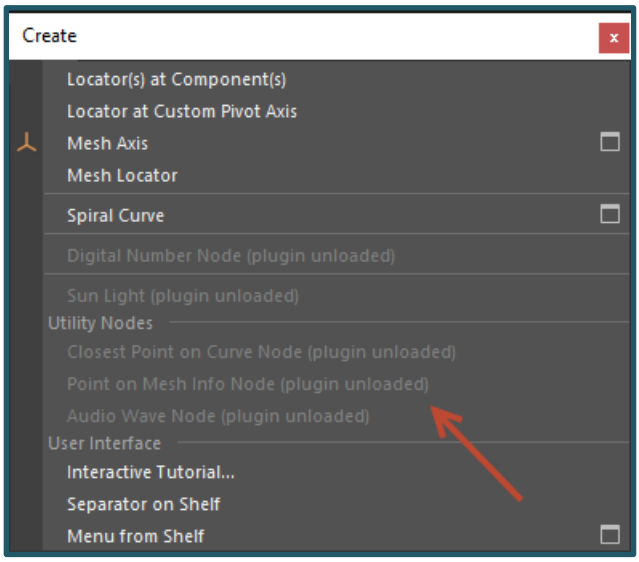
Installation

New in Maya 2025 you no longer need to download and install BonusTools separately. All of the BonusTools scripts and plugins are bundled with the regular Maya installer. You simply need to load the BonusTools plugin from the plugin manager and agree to the usage terms,

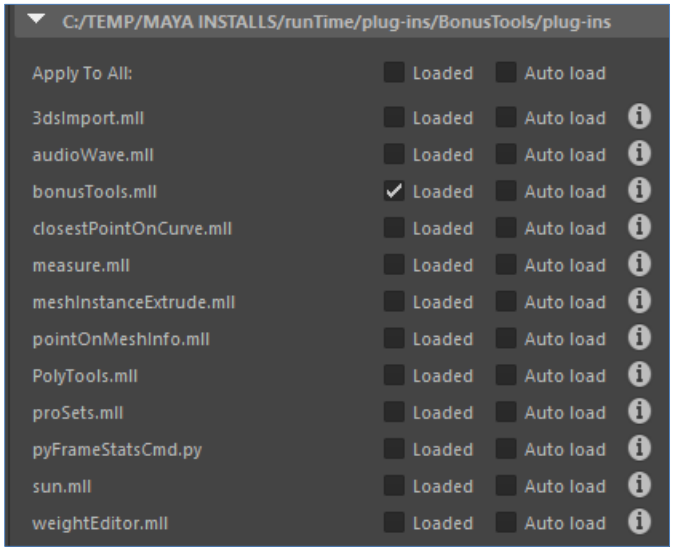


Plug-ins

Bonus Tools contains a number of plug-ins. Any tools that require one of these plug-ins will be greyed out in the menus by default and therefore unusable.



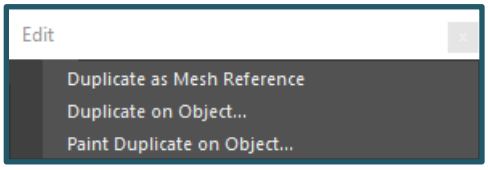
In order to use any of these tools you must first load the associated plug-in from the Plug-in Manager.



Be sure to set any of the Bonus Tools plug-ins that you wish to use in the future to auto load.

After loading one or more plugins, simply run (BonusTools -> Help -> RefreshBonusToolsMenu) and the menu will be updated to recognize any recently loaded plugins.

Edit



Duplicate as Mesh Reference

Select one or more polygon objects and apply. A duplicate will be created for each mesh as a "one directional" instance. This will allow you to make component or topological changes to the original object and see the same results on the duplicate. This is similar to an instance, however an instance works in both directions. In this case however, unlike an instance, if edits are made to the duplicate mesh they are layered on top of the edits of the original mesh, but the original mesh will not be affected. History can be deleted on the duplicate object in order to "bake" in the effect.

Note: This is similar to a feature called poly reference in 3ds Max.

Duplicate on Object

This tool, formerly called "Replicate Object on Components" has been relocated from the Modify menu.

This tool can be used to create copies or instances of a selected object at the location of either selected components or random components on a selected object.

The Method option will determine if it uses **selected** components or **random** components based on the second selected mesh.

The Duplicate option will determine whether **copies** or **instances** are created.

Similar to the **Snap Align Object to Component** tool it will snap and/or align the duplicate objects to the given verts/edges/faces. The duplicate objects will be moved to the center of each target component.

- Orient: Object will be rotated to align to the normal of the component. If the
- Constrain: Object will also be constrained at the point and to that normal.
- Parent: Object will be parented to the transform/object of the selected target component.

To use, select one transform followed by either a collection of poly components for **selected** method or a single poly mesh for **random** method, then apply **Replicate Object.**

Paint Duplicate on Object

This tool is an updated version of the legacy "Paint Geometry" tool. The old tool was buggy and unintuitive. If you are in still in need of the legacy tool, it is available in as an example script as part of the standard Maya installation in the (.\scripts\others) folder and you can still the tool with the following commands…

```
ScriptPaintTool;  artUserPaintCtx -e -tsc "geometryPaint" `currentCtx`;
```

The new version of this tool allows you to more easily interactively place duplicates of an object onto another objects by painting. Select the source object that you wish to paint first (e.g. a tree) followed by the target object that you want to paint onto (e.g. a terrain) then activate the tool. You will be placed into a paint mode and a UI will popup that will provide a few simple options.

You can choose between vertex snapping and UV snapping when placing. Vertices snapping will look at the actual vertices of the target object. If you need more granular placement you can use UV space snapping, but keep in mind that your target object will need to have clean, somewhat even UVs in order for this mode to work.

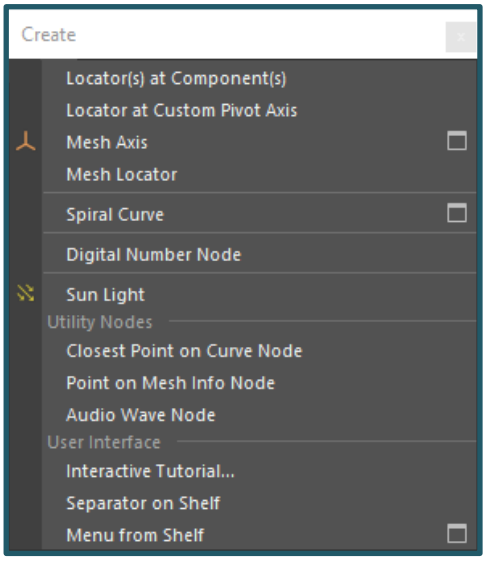
The create options determine how you the duplicates get created. If you choose instance each will be an instance rather than a standard copy. Align will cause the rotation of the duplicates to align to the surface of the target object, but it is assumed that Y is the up axis and the pivot of the target is at the appropriate location. Group will put all of the new duplicates under a single parent node.

The behavior of the tool can be changed with modifier keys. CTRL while painting will delete existing duplicate objects but only while the tool is active. If you quit and restart the tool again it will not delete previously placed objects from a previous use of the tool. Shift will modify either the rotation or the scale of the existing duplicates.

The modify options work with the Shift modifier. These are simple random settings that allow you to quickly randomize the rotation and/or scale of existing duplicates. These are hard coded to work between 0-360 for rotation, and between 0.25 and 1.75 for scale. More flexibility may be added in future versions.

Note that the source object can be just about any transform type, but the target must be a mesh or a surface.

Create



Locator(s) at Component(s)

This creates a new locator at the center point of every selected vertex, edge or face. If multiple faces, edges or vertices are selected then a separate locator will be create at each component.

To use select 1 or more vertices, edges, faces or mesh objects and click **Locator(s) at Component Center.**

Locator at Customer Pivot Axis

This creates a new locator at the current active pivot. If a custom pivot has been set the locator will inherit the custom rotation and position of that pivot.

To use select a mesh component or object, customize the pivot if desired, then click **Locator(s) at Pivot Axis.**

Mesh Axis

This creates a mesh object that represents the x, y and z axis. However, unlike a locator it has thickness that can be adjusted as well rendered. The resulting object is similar to a primitive in that it has some basic parameters that you can adjust including Scale, Radiu, Length, Subdivisions and Right/Left Handed. It can be especially useful for riggers to help visualize the axis orientation of a given control object.

To use simply click **Create Mesh Axis.**

Mesh Locator

This creates a mesh object that is visually similar to a locator. However, unlike a locator it has thickness that can be adjusted as well rendered. The resulting object is similar to a primitive in that it has some basic parameters that you can adjust to alter the **Length** and **Thickness** of the arms of the locator. However, it's not

a true primitive as it is using other existing nodes "under the hood" such as polyCube and extrudeFace to create the shape.

Note: The **Length** and **Thickness** attributes are on the transform node itself, which can be seen in the Channel Box or Attribute Editor. As will all Maya mesh objects the construction history can be deleted to "bake" in the effect.

Update: Precision for Length and Width has been increased with minimum values now set to 0.01

To use simply click **Create Mesh Locator.**

Spiral Curve

This creates a curve in the shape of a spring. The option box provides control over the height, radius and number of turns. Number of turns can be a negative number to have the twist in a different direction.

To use open **Create Spiral Curve** options and apply.

Circle Under Transform

This tool has been replaced by a more advanced version called **"Create Circle Control"** found in the **Rigging** menu.

Digital Number Node

This tool creates polygonal geometry in the shape of a digital readout for displaying values in the viewport

To use, run **Create Digital Number Node**. Select the PolyReadout node and then find the polyDigits node in the input section of the ChannelBox. This node will contain the various attributes for refining the look of the numbers.

Note: This requires the PolyTools.mll plug-in to be loaded - see BonusTools section of the Plug-in Manager.

Formatted Text Node

Create a node that converts input values into various formats (e.g. date or timecode) which display in the viewport as 3d text and displays in the viewport as 3d text

To use, run **Create Formatted Text Node**. By default, the node will have the current frame number as it's input, but any other value can be hooked up to the node by making a new connection to its input in the hyperGraph.

Note: This requires the stringFormatNode.mll plug-in to be loaded - see BonusTools section of the Plug-in Manager.

Sun Light

This creates a light that behaves like the sun. This is basically a directional light with additional controls. It allows the user to control attributes such as latitude, longitude, day of year and time in order to simulate the sun.

To use, run Create Sun Light Node. Select the light and look for the sunDirection node in the inputs of the ChannelBox. This node contains the various sun controls.

Note: This requires the sun.mll plug-in to be loaded - see BonusTools section of the Plug-in Manager

Closest Point On Curve Node

This plug-in defines both a MEL command and a DG node which takes in as input, a NURBS curve and a world space position, then computes the closest point on the input curve from the input position. In addition to the world space "position" at the closest point on the curve, also returned are the "normal", "tangent", "U-parameter" and "closest distance from the input position", at the closest point on the curve.

Update: This feature was inadvertently removed in 2022 but included again for 2025

Note: This requires the closestPointOnCurve.mll plug-in to be loaded - see BonusTools section of the Plug-in Manager.

Extended documentation for this tool can be found in the Bonus Tools install location ->

.\ MayaBonusTools-2025\Contents\docs\closestPointOnCurve.txt

Point On Mesh Info Node

This plug-in defines both a MEL command and a DG node which computes the world space position and normal on a poly mesh given a face index, a U-parameter and a V-parameter as input. This command returns information for a point on a poly mesh, in particular its world space "position" and/or "normal" at a given input face index, U-parameter and V-parameter.

Update: This feature was inadvertently removed in 2022 but included again for 2025

Note: This requires the pointOnMeshInfo.mll plug-in to be loaded - see BonusTools section of the Plug-in Manager.

Extended documentation for this tool can be found in the Bonus Tools install location ->

.\ MayaBonusTools-2025\Contents\docs\ pointOnMeshInfo.txt

Audio Wave Node

This is a command to create a node which will output the amplitude of a sound node.
To use:

- Load a sound file into Maya.
- Bonus Tools -> **Create Audio Wave Node**
- Enter one of the following MEL commands:
    - setAttr audioWave1.audio -type "string" "soundfile";
    - connectAttr soundNode.filename audioWave1.audio;
- Enter the following MEL command:
    - connectAttr time1.outTime audioWave1.input;
- Enter the following MEL command:
    - connectAttr audioWave1.output object.attribute;

The audioWave node has two attributes which let you scale the sampled amplitude:

- sample - this attribute specifies how many sound samples to average to compute amplitude ... the bigger this number is the longer it will take for a drastic change in amplitude to be detected
- scale - this attribute is a scaling factor for the resulting amplitude with a scale of 1.0 the amplitude will fall within a range of -1 to +1

Interactive Tutorial

Use this tool to create interactive tutorials through an easy-to-use user interface. With it, you can create steps, edit and place text, automate actions (via python code), and monitor for specific trigger events. The app will handle creating and connecting the appropriate nodes behind the scenes for you.

Release notes:

V 2.0 (Compatibility note!)

- Added the ability to place overlays relative to objects in the Viewport.

- In previous versions, OS font scaling would displace word bubbles, causing the "tail" to no longer point at the same spot. To fix that, this version of the script places all bubbles relative to the tail rather than by the top-left corner. Now, regardless of the font size, the tail will always be in the same spot.

- Compatibility branch: Due to the changes above, this script is incompatible with tutorials created by previous versions of the script.

V 1.2

- Fixed a bug that broke the "Import curent camera + values" button when the camera had a specific kind of name

- Changed the default title of the controller to the corresponding stage name, and the description to read "Press Next"

- Fixed a bug that caused the controller to stay hidden when using "Play from Selected".

V 1.1

- Fixed a bug that broke the "Import current camera + values" button on the Initialize stage component

- Fixed a bug that caused text overlay previews to always appear centered when "Relative To" was set to "UI Code". (Note the actual placement was always working, just the preview was broken)

- Fixed a bug that caused an endless loop in the background if two Time Slider bookmarks were created in sequence without the tutorial being played through between. (Note the original bug did not prevent user interaction and could be fixed by simply running the tutorial once)

V 1.0

- First release

To create a new tutorial with this app:

1. Start with a new (empty) Maya scene.

2. Load the Tutorial Creator app from the Maya Bonus Tools.

3. In the Tutorial Creator window, click the "Create new tutorial" button.

The app will automatically create the tutorial back end and initialization step (AKA: the tutorial's welcome screen).

The Tutorial Creator window is divided into a number of sections.

Clockwise from top-left they are:

1. New Step button – Click this to add a step to the tutorial. When creating a step, you can also specify where the step should go (default to the end of the current set of steps, but you can also choose to insert a step between existing ones), as well as what components a step should include (text, triggers, etc). Don't worry, you can always add more components after you've created a step (see "Step Components" below).

2. Play buttons – Click the appropriate play button to start the tutorial from the beginning, or from a specific selected step in the Node List (see 4).

3. Editor Panel – This is where you can customize the behavior of the tutorial. The Editor Panel will change depending on the node you have selected in the Node List (see 4).

4. Node List – Displays a list of all relevant tutorial nodes connected to Stage0 (the start of your tutorial). You can select nodes in here to manipulate their behavior via the Editor Panel (see 3). You can also rename or delete nodes via the two buttons under the Node List.

- Stage nodes represent each step of the tutorial. Options for stage nodes include: adding a new component (see Step Components), auto-playing the Time Slider when a step begins, moving to the next step when the animation finishes or immediately or after a certain time delay, and creating / linking a Time Slider Bookmark to the step to immediately jump to that frame range when the step begins.

- Step Components are indented under stage nodes and represent the support nodes that determine what happens at each step (see Step Components below).

Step Components

For each stage node in your tutorial, you may have an arbitrary number of step components associated with it. The four types of step components are:

1. Text Overlay: Display and position text on screen. This can take the form of either a non-interactable word bubble, or a fully movable dialog box.

2. Controller Update: Update the tutorial controller. Updates may include changing the text, locking buttons, or incrementing the step counter.

3. Stage Setup: automate things to happen in Maya before a step begins. For example, you may want to show or hide the Attribute Editor or Channel box, change the current active camera, or even execute entire Python strings for more complex setup automations.

4. Trigger: Define a trigger to execute code when a specific event occurs. The default setting will move the tutorial to the next step, but you can customize the behavior to do whatever you want. For example, you may want to monitor for the user to select a specific tool, then display some additional text.

Additional Options

In the Tutorial Creator windows menu bar, you will find some additional options. These are:

- Jump to selected stage time: Enable this if you want Maya to automatically set the Time Slider range to appropriate Time Slider bookmark when you select nodes in the Node List. Note: If you select a stage (or step component) without an accompanying bookmark, then Maya will jump to the nearest applicable bookmark from the stages before the selected one.

- Jump to selected stage camera: Enable this if you want Maya to automatically set the current camera to that of the selected stage (or step component) when you select nodes in the Node List.

- Source text from dictionary: Allows you to bypass the "text_overlay" or "update_controller" components and instead source text from a single dictionary script node (populateText). This is mostly useful if you want to centralize the text for localization purposes.

- Show in Node Editor: Select this to show the tutorial nodes in Maya's Node Editor. Useful for advanced manipulation of the nodes.

- Show in Expression Editor: Select this to display the Step Components in the Expression Editor. Useful for advanced manipulation of the Python scripts.

- Utilities: Several miscellaneous tools that you may find useful when creating tutorials, including: Showing your current selection, showing a list of contexts, and merging text from the "text_overlay" and "update_controller" nodes to the aforementioned dictionary script node (populateText). There are also options to export a json file for localization, as well as importing the translations to alternate languages of the tutorial and overriding the current Maya language (for testing localized scripts).

Separator on Shelf

Beginning in Maya 2016 you can better organize shelves with visual separators. However, this feature is only available through a MEL command. This tool will create a separator on the shelf that is currently active. You can create as many separators on each shelf as you need. Once created you can then use MMB to move shelf items in-between the separators. They can also be easily deleted via the Shelf Editor.

To use simply select the shelf that that you want the separator to go on and click Create Separator on Shelf.

Menu from Shelf

This tool will convert a user specified shelf into a pull-down menu that will be added at the end of the top Main Menu. The menu will only exist during the current Maya session but will disappear if Maya is restarted.

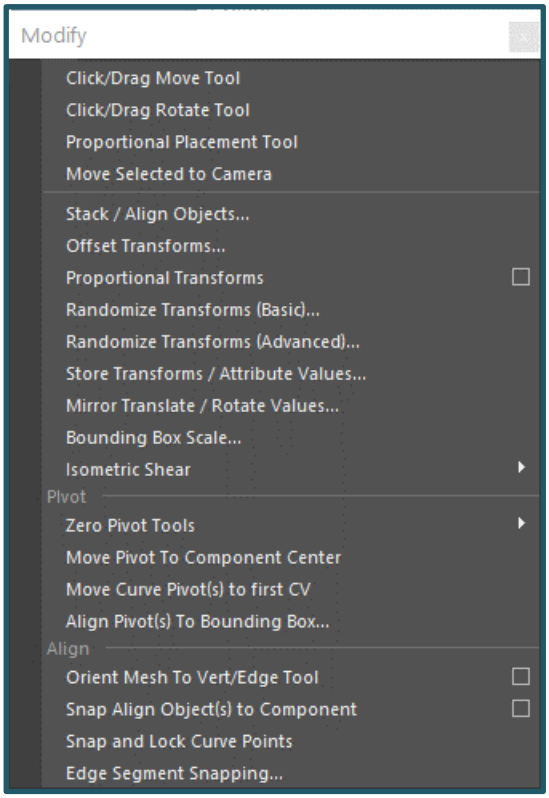
Optionally, the menu can be written to a MEL file that can be reused. When writing the file to disk, the menu can later be added to the userSetup.mel file so that it is will always be available at start-up. The UI/menu name will always be of the form main_*ShelfName*_Menu and the saved file will be called *ShelfName*_Menu.mel and will be save in your user scripts folder.

The save feature was previously broken but has been fixed. However it should be noted that the resulting (.mel) file will only work for shelves (i.e. shelf buttons) containing standard commands and/or procedure names. Shelves containing entire scripts will likely cause errors in the resulting MEL file that may be difficult to track down.

To use open the **Create Menu from Shelf** options and type in the name of the shelf that you would like to convert into a menu. The default is custom but any shelf can be chosen. You will need to make sure that the desired/named shelf is also the currently active shelf.

To delete a menu you will need to use the deleteUI command or simply restart Maya. If you created a menu called MyTestShelf, you would execute this MEL command -> deleteUI main_MyTestShelf_Menu;

Modify



Click/Drag Move Tool

This tool can be used to interactively place and drag an object or multiple objects along a virtual ground plane. Simply activate the tool, and then click drag in the viewport to move the selected object(s) around on the current XZ plane. CTRL click to drag vertically along Y. The Y value of the object(s) will determine the virtual ground plane. The tool can also be used to quickly place object at an location from the Top camera view.

Note: creation of <ctrl+shift+W> hotkey for easy access.

To use, select one or more objects/transforms, activate the **Click/Drag Move Tool**, and then click/drag in any viewport. Use ctrl+shift+W to reactivate tool quickly.

Click/Drag Rotate Tool

This tool can be used to interactively rotate an object or multiple objects along their vertical axis. Y up is the default but Z up is also supported. Simply activate the tool, and then click drag in the viewport to rotate the selected object(s) around on the Y axis. If multiple objects are selected, then a shared center is used based on the overall bounding box. CTRL click to drag rotate around last selected object (e.g. a locator). Shift click to drag rotate around each object's individual pivot.

Note: creation of <ctrl+shift+E> hotkey for easy access.

To use, select one or more objects/transforms, activate the **Click/Drag Rotate Tool**, and then click/drag in any viewport. Use ctrl+shift+E to reactivate tool quickly.

Proportional Placement Tool

This tool will allow you to interactively move, rotate or offset several objects in the viewport without having to group them together. The Universal Manipulator is used to interactively set position, orientation and spacing of selected objects. Use the translate handles to position, the rotate handles to orient and the scale handles to offset existing spacing in a proportional way. Note that the objects themselves will not be scaled but instead the spacing between them will be. To achieve this a series of temporary locators and constraints will be created while the tool is active, but these will be automatically deleted as soon as the selection changes.

To use select two or more objects and activate, then interactively move, rotate or offset objects in viewport using manipulator.

Move Selected to Camera

Select one or more objects and apply. All objects will be moved and positioned directly in front of the camera associated with the active panel.

To use select on object and apply

Stack / Align Objects

This tool will either stack or align selected objects along a given axis based on their bounding box values. The last selected object will be the "target" object and all other selected objects will align relative to the target.

Use Stack Above, Stack Below and Align Along for aligning objects along the desired axis.

You can also align to the min, mid or max bounding box values of X, Y or Z.

The Equidistant method will equally space all selected objects within the min and max of the collective bounding box and along the chosen axis.

Offset Transforms

This tool will quickly offset translate, rotate and scale values for selected objects. Set amount of offset for move/rotate or percentage of offset for scale. Move and rotate can work in local or world space.

Proportional Transforms

This tool will transform selected objects in a proportional manner. A percentage value is specified to define how much each object will be affected based on their current values as well as a pivot location. You can Translate, Rotate or Scale and have the offset proportionally affect each object based on either the last objects pivot or a user defined pivot location.

An option for using the first selected object as the pivots location has been added.

To use, select the object you wish to transform and open the **Proportional Transform Window**.

Randomize Transforms - Basic

This tool will quickly randomize translate, rotate and scale values for selected objects. Set min and max values for range of randomization. Values can be randomized on one, two or all three axis.

Randomize Transforms - Advanced

Randomizer lets you interactively apply random values to selected objects, components and channels.

Note: This is a python based tool and may not work after default installation. See installation instructions on setting up your python path if you want to use this tool.

To use, select some objects and start the Randomizer. (or open Randomizer, then select objects and click on Reload Sel.) A window will open with sections for:

Objects, Channels, Components, Selection

Objects: Simple, easy to use randomization for translate, rotate and scale.

- Move the **Range** slider to apply random settings (might be slow for large >500 number of objects).
- Amplify or reduce the randomization by using **Multiply XYZ**.
- Use Maya´s Move/Rotate/Scale tool to apply offsets.

There is no **Apply** button. Click on **Reset** to remove all randomization.

To keep the current settings simply close the Randomizer window or click on **Reload Sel**. Settings will be applied and all values in the Randomizer window will be reset. The current selection gets loaded.

Channels:

- To randomize single channels, select the channels in the Channel Box and click on **Reload Sel** in the Randomizer window.
- Use **Offset** to apply an offset.
- Set **Step** to apply the random numbers in steps (**Offset** is independent from **Step**).

Components: Select vertices and click on **Reload Sel**.

- Apply a **Translate Range** and **Scale Axis** as in the **Objects** section. All vertices move independently.
- Use **Dnoise** to apply a 3D fractal function as random translation. The **Scale Dnoise** sets the scale of the Dnoise fractal.
- Note: The **Along Normal** setting is currently not working.

Selection: Use the **Selection Ratio** slider to select a ratio of the loaded selection. Note: The ratio is only the probability that an object gets selected. There is no guaranty that the ratio is met.

Buttons:

- **New Seed**: Creates a new seed of random numbers. Use this button if you don´t like the current randomness.
- **Reload Sel**: Leaves all objects where they are, reloads the current selection, and resets all values in the Randomizer window.
- **Reset**: Resets all values in the Randomizer window to default. Selected (loaded) objects will reset to their initial settings.

Undo: Avoid to using Undo while the Randomizer window is open. An Undo step is set whenever a slider is released.

Store Transform / Attribute Values

This tool will allow you to save the current values for the selected object(s) as a shelf button that can be used later to reset to object(s) to those stored values. This is similar in concept to a preset. There is also an option to include hierarchies as well as underlying shape nodes for things like cameras and lights. Any attributes on the transform or shape node will also be stored.

Note: This tool supports custom (i.e. dynamic) attributes. This will allow you to now store settings on nodes that contain user defined attributes for a number of purposes. The most common example would be storing values on a custom rig or control objects. So if you have a control node driving the poses of a hand (e.g. Open Hand or Spread Fingers) you can now store those poses on a shelf.

NOTE: Formerly in the Animation menu and known as "**Store Pose to Shelf**" this tool can be used to record a given pose for a character by creating a shelf button and icon on the current (top) shelf. This button contains all of the necessary setAttrs for getting back to a particular pose. It will work on either hierarchies or selected nodes. You can create as many poses/buttons as you need. This pose is not to be confused with Trax Poses. There is no association between these poses and Maya's Character Sets.

To store a pose for an entire character, select the parent node(s) of anything related to the character (Root Joint, IK, constraints, locators, etc...) If everything is parented under one node, simple select that node. Then just open Store Pose to Shelf, name the button and apply using hierarchy mode. To return to the stored pose, simply click the shelf button.

This tool does some basic error checking, including checks for attributes that are un-writable or locked. (e.g. EndEffectors or IK handles that are constrained or driven by another object)

To use, select one or more objects and open **Store Transform Values**. Then press either the store selected or store hierarchy button in order to create a shelf button on the current/active shelf. You can also choose the name of the button before creating it.

Mirror Translate/Rotate Values

This tool will invert/mirror the Translate and/or Rotate values on all selected objects across a chosen axis.

To use, select one or more transforms and open **Mirror Translate/Rotate Values**. Choose the transforms to mirror as well as the axis (x, y, z) and then apply.

Bounding Box Scale Window

This window allows you to scale an object based on its global bounding box size. The scale pivot of the object can be easily set to the minimum or maximum X/Y/Z position of the bounding box. This essentially allows you to scale an object independent of the actual scale values in order to fit the object into arbitrary world dimensions.

Update: Improved UI layout and usability, auto display of scale pivot, better feedback in command prompt

To use, select an object to scale and open the **Bounding Box Scale Window...**

Isometric Shear

Update: This menu, formerly called "Isometric Render" has been relocated from the Render menu. The auto render option has been removed as it relied on the legacy Maya renderer. This tool is more related to transformation, so it was relocated to the modify menu. You can still easily render the result of the Isometric World Node with any renderer.

> **Create Isometric World Node:** All transforms in the scene will be grouped under a new node and the shear attributes for that node are exposed and set to create an isometric render effect. You can adjust the shear attributes on the IsometricWorldNode to alter the affect for different points of view.
>
> **Remove Isometric World Node:** Resets the scene so it is not in isometric mode, by un-parenting all transforms from the IsometricWorldNode.

Zero Pivot Tools:

Keep Pivot Offset and Zero Local Values

This tool will maintain the existing pivot offset but will modify the underlying values so that the local pivot translate values will all be zero (0,0,0). This means that if you type in 0, 0, 0 for the object's tx, ty and tz, then the object will move with its pivot to the origin as you would expect.

This can be useful when placing and snapping objects when translate has been frozen with an offset.

Center Pivot and Zero Local Values

This tool will move the selected object(s) pivot to its own center, but will modify the underlying values so that the local pivot translate values will all be zero (0,0,0). This means that if you type in 0, 0, 0 for the object's tx, ty and tz, then the object will move with its pivot to the origin as you would expect.

This can be useful when placing and snapping objects when translate has been frozen with an offset, but also can be helpful when setting up objects for rigid dynamics (e.g. Bullet)

Move Pivot to Base and Zero Local Values

This tool will move the selected object(s) pivot to its own base (i.e. bottom Y and center X/Z) but will modify the underlying values so that the local pivot translate values will all be zero (0,0,0). This means that if you type in 0, 0, 0 for the object's tx, ty and tz, then the object will move with its pivot to the origin as you would expect.

This can be useful when placing and snapping objects when translate has been frozen with an offset.

Move Pivot to Origin and Zero All Values

This tool will move the objects pivot the world origin, but will modify the underlying values so that the local pivot translate values will all be zero (0,0,0). It will also zero out the local rotation axis values. This means that if you type in 0, 0, 0 for the object's tx, ty and tz, then the object will move with its pivot to the origin as you would expect.

This can be useful when placing and snapping objects when translate has been frozen with an offset.

Move Pivot to Component Center

This tool will move the object's pivot point at the center of the selected component(s).

To use, select one or more polygon components and apply **Move Object Pivot to Component Center**.

Move Curve Pivot(s) to first CV

This tool will move the pivot for one or more selected NURBS curve to the first UV of each curve. This will allow you to scale or rotate relative to the origin of the curve. This can be particular useful when working with curves for XGen and nHair. For example, you may want to build curves that can be used as guide curves, or more easily work with nHair output curves as in both cases the curve pivots will default to the origin.

To use, select one or more transforms and apply **Move Curve Pivot(s) to first CV**.

Align Pivot(s) to Bounding Box

This tool will align the pivots of all selected objects to the bounding box of either each object individually, or to the bounding box of all objects combined. The center pivot first option will put the pivot at the center of the bounding box before setting it to the min/max of the chosen axis. Leaving this option off will cause the pivot to line up relative to its previous location instead.

To use, select one or more transforms and apply **Align Pivot(s) to Bounding Box**.

Orient to Vert/Edge Tool

This tool orients one mesh to another mesh by aligning a vertex from the first object to a vertex or an edge on the second object. The orientation can take place around any single axis: X, Y or Z and can occur in Object or World space.

To use, activate the **Orient to Vert/Edge Tool**, set the "orient to" mode in the options to either edge or vert. If edge is selected, then you will click on a source vert from the first object and a target edge on the second object.

If vert is selected, you will click on a source vert from the first object and a target vert on the second object. The result will be a rotational alignment of the first objects to the second. The location of the pivot will also affect the rotation.

Snap Align Object(s) to Component

This tool can be used to snap and/or align a selected object to a selected (target) poly component. The target can be a vertex, edge or face. The object will be moved to the center of the component. This tool can snap/align multiple object at a time but only to one component at a time.

- Orient: Object will be rotated to align to the normal of the component. If the
- Constrain: Object will also be constrained at the point and to that normal.
- Parent: Object will be parented to the transform/object of the selected target component.

To use, select a one or more transform/object, followed by a mesh component (vert, edge, face), then click **SnapAlign Object**

Snap and Lock Curve Points

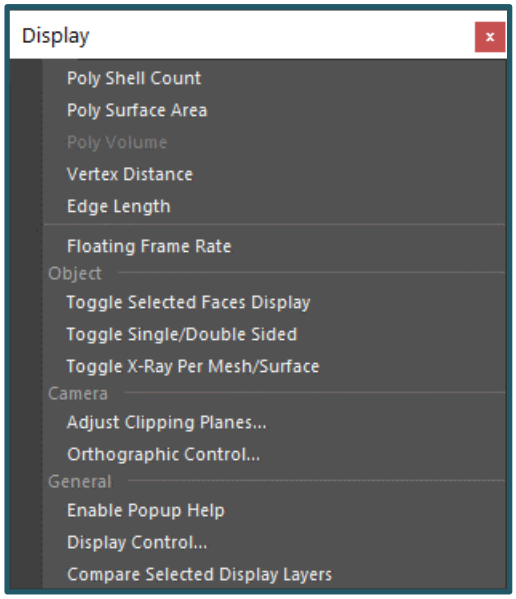
Formerly known as "Curve Intersection Lock Node" this tool locks two curve points or two curve edit points with a locator. After the points are locked, the locator can be moved to change the shape of the curves. If the curves are moved the locator will pin the target points and keep them in place.

To use, select a combination of curve points or edit points and then apply **Snap and Lock Curve Points**

Edge Segment Snapping

Use this tool to quickly and easily enable edge segment snapping so that you can snap objects and components to divisions of a given edge. You have the option of snapping to edge centers, thirds, quarter or fifths. When a pull-down menu is selected the edge/curve snapping toggle will be enabled and the appropriate magnet and tolerance settings will be applied. If you need more divisions/segments there is also a shortcut provided to the global user prefs where you can set the values manually.

Display



Poly Shell Count

This item will add Heads Up Display to show the total number of distinct shells within one or more selected poly objects. The menu item is a toggle.

To use, run **Toggle Poly Shell Count** to turn it on. The HUD value will dynamically update any time you select one or more poly objects.

Poly Surface Area

This item will add Heads Up Display to show the total surface area of the selected mesh. If more than one mesh is selected it will calculate the combined total. The menu item is a toggle. This was previously just a command that printed in the output window but been converted to a dynamic HUD.

Note: This used to require a plugin but no longer does.

To use, run **Toggle Poly Surface Area** to turn it on. The HUD value will dynamically update any time you select one or more edges.

Poly Volume

Update: Fixed load error // line 1: Cannot find procedure "measureCreateUI". //

This item will add Heads Up Display to show the total volume of the selected mesh. If more than one mesh is selected it will calculate the combined total. The menu item is a toggle. This was previously just a command that printed in the output window but been converted to a dynamic HUD.

Note: This requires the measure.mll plug-in to be loaded - see BonusTools section of the Plug-in Manager.

To use, run **Toggle Poly Volume** to turn it on. The HUD value will dynamically update any time you select one or more edges.

Vertex Distance

This item will add Heads Up Display to show the distance between two selected poly vertices. The menu item is a toggle.

To use, run **Toggle Vertex Distance** to turn it on. The HUD value will dynamically update any time you select two vertices.

Edge Length

This item will add Heads Up Display to show the length of a selected poly edge. If more than one edge is selected it will calculate the total length of all of the selected edges. The menu item is a toggle.

To use, run **Toggle Edge Length** to turn it on. The HUD value will dynamically update any time you select one or more edges.

Floating Frame Rate

This item will create a floating, movable Heads Up Display for dynamically showing the current frame rate. It uses a simple annotation node which in turn uses a callback to get the value of the standard frame rate HUD. It can be enabled and disabled as a toggle. The HUD / annotation is a transform, which means it can be moved anywhere within the scene so that it can be clearly seen from the active camera. It can also be parented under another transform so that it follows a particular object.

Update: Color has been changed to white for better visibility

Note: If you delete the node without disabling it, the callback will continue to run and print errors in the Script Editor output. Simply run again to disable it before deleting, or open an new scene and it will be automatically disabled.

To use, run **Floating Frame Rate** to turn it on. Apply again to disable it.

Toggle Selected Faces Display

This tool will show or hide selected polygon faces and/or objects. A Heads-up Display button will be created in the viewport panels that will allow you to either hide additional faces or show any faces that are hidden. This essentially works in the opposite manner as the standard isolate select.

To use, select some faces and/or objects and run **Toggle Selected Faces Display**. They will be hidden in all viewport panels. After running once, a Heads Up Display will be created that can be used for additional hiding/showing.

Toggle Single/Double Sided

Toggle Single/Double sided display of selected objects. If nothing is selected, then toggle will be applied for all mesh objects in the scene.

Note: for multiple objects you may need to apply twice in order to sync up display for every mesh.

Toggle X-Ray Per Mesh/Surface

Toggle XRay display mode on a per mesh/surface basis. Select one or more meshes or surfaces and apply in order to enable or disable to per object XRay setting. This option is supported in both the legacy viewport as well as viewport 2.0.

Note: for multiple objects you may need to apply twice in order to sync up XRay display for every mesh.

Adjust Clipping Planes

This tool provides a simple window with slider that will dynamically update the near and far clipping planes for whichever view is active. Note: A panel with a camera must be active in order to open the window.

To use, click in any camera view in order to make the panel active. Then run **Adjust Clipping Planes** and adjust sliders to change that camera's clipping planes interactively.

Orthographic Control

This tool creates a set of Heads Up Display buttons directly in the viewport that can be used to work with Orthographic cameras.

- **Lock / Unlock:** Use this to lock and unlock the tumble / orbit of the camera which is typically locked by default for orthographic cameras.
- **Frame:** Use this to either frame to selected objects or if nothing is selected frame to the origin. Unlike the standard frame with the "f" hotkey, this method will ensure that the center of interest of the camera is set appropriately, so that you can in turn tumble around a given object predictably.
- **Reset:** Use this to set the active camera back to a flat, orthographic point of view.

Display Control

This tool will create a simple set of Heads Up Display buttons and slider in the lower left corner of each viewport. The buttons can be used to control various display settings for your scene and objects in your scene. There are three groups of buttons:

- **View Display**: control things like UI, Camera and Grid settings
- **Anim Display**: control display of things like Deformer, IK and Joint settings

- **Poly Display:** control display of things like Culling, Normals, Borders, CPV, and more.

You can use the "**minimize**" button to access the other groups or you can use it to simplify the UI when not using it.

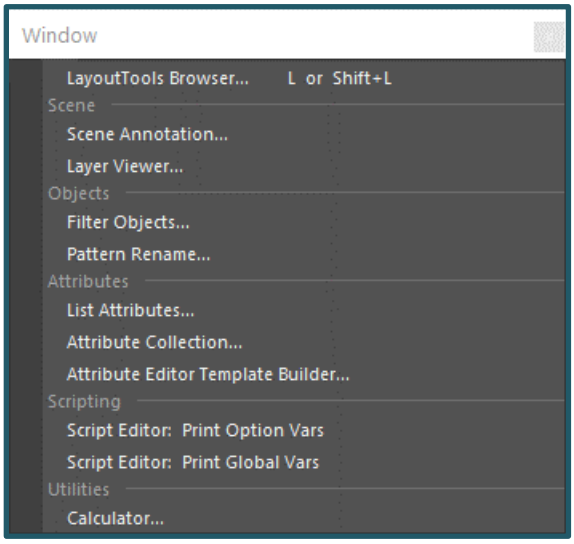
Note: Auto creation of <ctrl+2> hotkey for easy toggling on/off of HUD. This makes it very easy to access many different display setting that are otherwise scattered all around the Maya UI. A number of bug fixes are included as well.

Compare Selected Display Layers

This tool can be used to compare the contents of two display layers by alternating / toggling the visibility setting of each. If both are visible, then one will be made invisible. Applying again will flip the visibility between the two. You can hit the "g" hotkey to keep toggling back and forth between the 2 layers and visually comparing the contents of each.

To use, select 2 and only 2 display layers form the Layer Editor and apply **Compare Selected Display Layers**

Window



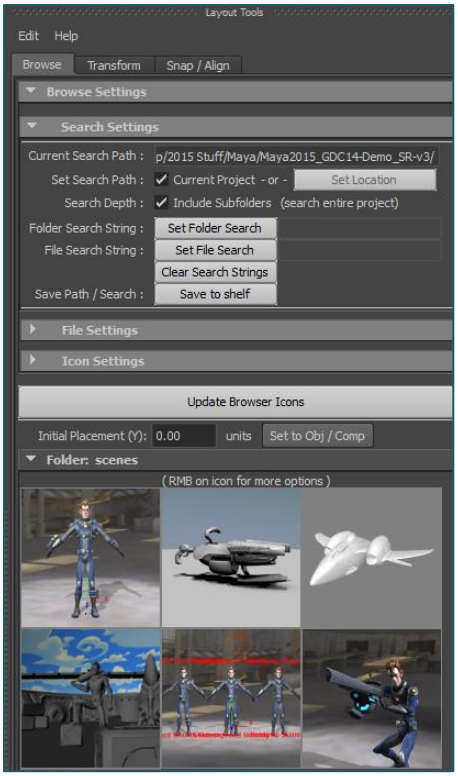
LayoutTools Browser

LayoutTools provides new functionality and also consolidates many of Maya's existing features into one streamlined, easy to use interface. The interface itself is divided into 3 main sections: **Browse, Transform** and **Snap/Align**.

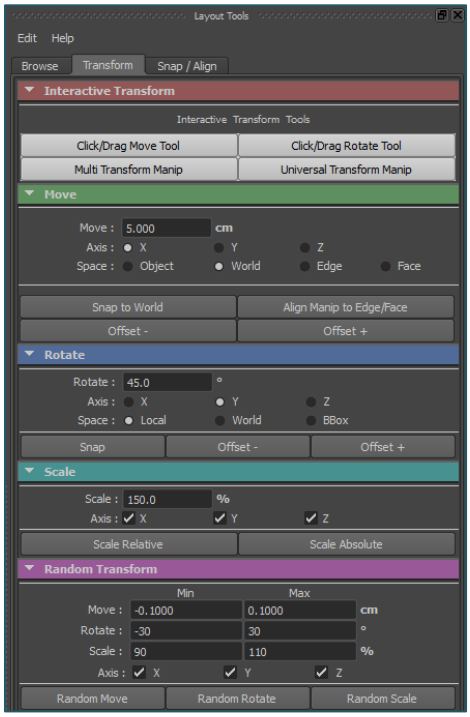
Note: Auto creation of L and Shift+L hotkey for launching LayoutTools can be setup when you first launch the LayoutTools UI

Update: Several bugs have been fixed in this version including crashes that were occurring when updating or refreshing the icons in the file browser or when resetting options. Mental Ray has been removed as an option for icon creation.

Browse: This is now a general-purpose icon-based Maya file browser. It provides a simple icon-based UI, which allows you to quickly and easily browse any project or any other user defined location for Maya and FBX scene files. **LayoutTools** will search for applicable files and automatically build icon buttons for each file it finds. By simply clicking on an icon, the associated file can be opened. Alternatively, it can also be imported, referenced or assembly referenced into the current scene. If imported/referenced the object from within the scene will then be automatically selected so that it can easily positioned within the scene, simply by clicking in the viewport and dragging it into place. A new RMB menu for each icon provides quick and easy access to various options for dealing with the file including creating icons, as well locating the file via a native os browser.



Transform: contains tools for adjusting the position, rotation and scale of objects within a scene. This includes the ability to control the absolute or relative values of objects in various ways, including random, mirrored, and proportional transformations.



Snap/Align: contains a collection of tools for the alignment and snapping of objects within a scene. Standard Maya features such as the AlignTool and the SnapTogetherTool are integrated, as well as new tools such as Bounding Box Alignment and Target Snapping.

Note: The old **Display** section has been retired and its functionality has been streamlined and rolled into the new and separate BonusTool called the **Display Control HUD**

NOTE: More detailed info about LayoutTools can be found on the LayoutTools resource page. There you will find tutorials and updated documentation. Check out the Layout Tools resource page on the Area -> Layout Tools Resource Page

Scene Annotation

Update: Improved UI with adjustable width. Fixed major bug that was causing consistent crashes when closing scene annotation window.

This is a floating UI for the purpose creating and/or viewing annotations / notes that will save with a given scene and will be displayed again when the Scene is opened. This can be useful for sharing information and details about a given scene between multiple people or simply reminding yourself about something the next time you work with the scene again.

A warning dialog box has been added to prevent accidental deletion of existing Scene Annotation Nodes.

To use, open **Create Scene Annotation**, enter text into the main field and use file menu to save.

Layer Viewer

This is a floating layer editor with a viewport attached to it. The layers are displayed in an Outliner view so that they can be selected directly from the window. Once a layer is selected the contents of that layer will automatically be isolated and framed in the viewer window. The individual items in a given layer can also be

isolated by expanding the layer in the Outliner. The List All Layers button will then expose all layers allowing you to isolate a different layer. There are additional controls along the top for editing the selected layer and changing various display settings.

To use, open a scene with layers and launch the Layer Viewer Window.

Filter Objects

This opens a window which will filter and select objects in the scene. You can either filter by node type (e.g. locator or spotlight) or you can filter by classification (e.g. geometry, lights or shapes.) After you select a certain filter in the left hand column it will search the scene for nodes or objects of that type and list them in the right column. You can also select, delete and create sets for the nodes/objects that are listed.

To use, open the **Filter Objects Window** and choose the filter type in the menu.

Pattern Rename

Pattern Rename is a renaming tool that is useful for renaming extremely long object names that contain many equal patterns (for example imports from CAD systems). The script lets you enter a number of search strings (expressions) and optional replace strings and shows the resulting object names in a preview list.

The window of Pattern Rename has an input section (**Max Iterations**, **Prefix**, **Search**, **Replace**), a list of **Search and Replace Strings** with an **Add** and **Remove** button and a list of **Resulting Names**. At the bottom of the window you find buttons for **Rename**, **Reload**, **Refresh** and **Close**.

Usage: To start, select a bunch of objects (preferably with long names and many patterns in them) and click on **Reload**. The object names are then listed in the **Resulting Names** list.

Find a pattern in the names that you want to remove/replace and type that pattern into the **Search** field. (optional: enter a **Replace** string). Hit Enter or advance to the next input field to update the list of **Resulting Names**. This shows now a *preview* of resulting names.

To store/register your search/replace expression click on **Add**. This will add your entry to the list.

*(NOTE: The **Search** string is a regular expression. Please refer to the documentation of the MEL command "match" for a description of the syntax of regular expressions)*

To edit an expression in the list, simply click on it, make your changes in the input fields and click on **Add** to store your changes. To remove expressions simply select them and hit the **Remove** button (no undo here).

IMPORTANT: Only those entries that appear in the list of **Search and Replace Strings** will be used for the final rename. Temporary (unsaved) entries in the input fields will NOT be used for the final rename.

Prefix simply adds a prefix to all object names.

Max Iterations stops the replace loop after the number of iterations to prevent endless renaming (e.g. rename "Hello" with "HelloPLUS" and do that again/again/again...). If **Max Iterations** is set to 1 then only one occurrence of all search expressions gets replaced.

Sometimes the script warns you that the resulting names are invalid because they contain spaces or have a number at the beginning. Make sure that the names are ok before you click on the **Rename** button, otherwise the rename will produce errors.

The **Rename** button will rename all objects that were loaded into the **Resulting Names** list.

Only those objects that you have loaded with **Reload** or that were selected when you opened the window are in the list. At any time, you can change your selection and **Reload** it (e.g. load only a small number of objects for testing, then load the final selection right before you click on **Rename**).

List Attributes

This item lists all attributes for the selected object in a single scrollable column and provides information about the type of each attribute.

To use, select an object open **List Attributes**. The window will update dynamically if a new object is selected.

Attribute Collection

Update:

2.09 : Fixed the Edit Window "grow/shrink" problem

2.10 : Fixed the width of group buttons and buttonRow. Fixed the width problem with colorSliders. Added a new Type "AttrVector" which is a float compound for vectors. Added a "Default Height" value for all controls to the prefs

Attribute Collection (**V2.10**) is a tool that helps you create your own user interface (UI). It lets you select channels (attributes) in the **Channel Box** and creates fields and sliders for these channels in a new panel or window.

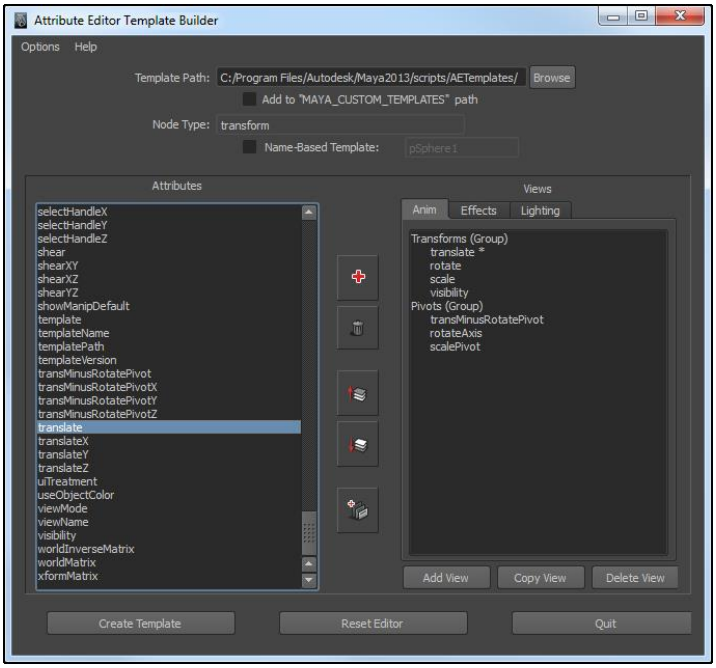
You can then interactively:

- sort, rename, regroup and delete fields/sliders
- set min/max values for sliders
- insert UI elements like separators, buttons, icons, text, etc.
- use these fields/sliders/buttons to easily adjust and keyframe your objects
- save the UI with the scene... when you load that scene the UI will open automatically.

Extended documentation for this tool can be found in the Bonus Tools install location ->

.\ MayaBonusTools-2025\Contents\docs\attributeCollection_2.03

Attribute Editor Template Builder



Introduction - With Custom Attribute Editor Attribute Views, you may use template files to customize the Attribute Editor for specific tasks, projects or artist type. Some things you can do with custom Attribute Views in Maya are:

- Create custom views of attributes in the Attribute Editor.
- Specify views for both Node Types and for specifically Named Nodes,
- Set views for the current selection, or Globally for all nodes in Maya
- Use Custom Callbacks in Python and Mel to inject UI or Actual Script Functions into the AE
- Set Maya to search for Templates in a custom director or across a network

There are 5 short video tutorials to get you going on View creation with the Template Builder. The Quickstart movie will get you off and running, but the other videos will open you up to the powerful potential of Attribute Editor views. These movies will be posted on the -> Bonus Tools Resource Page

Note that Pop-Up help is available for the Button in the Template Builder UI.

Basic Operation - To start the UI you either need to have a node selected that is of the type you want to create a template for before initiating the Template Builder, or you can select a node press the Reset Editor button to re-initiate the UI.

Several items in the Template Builder are automatically filled in, such as the Template Save path and the Node Type you will be creating a template for. The Attributes list will also fill in with a list of all the attributes for the selected node, and the Views list will contain three empty start-up views – Anim, Effects and Lighting.

Building Basic Node Type Views - Building views is a very simple process. Merely select attributes in the Attributes list and press the "+" button to add them to the current view. You can remove attributes from the current view by selecting the attribute(s) in current Views list and press the "Garbage Can" button to remove them from the current view.

With the "Move Up" and "Move Down" buttons you can shift attributes up or down in the current view. The order in which they are displayed in the Template Builder UI is how they will show up in the Attribute Editor when that view is displayed.

To add an Attribute Group for the purpose of visual organization, simply press the "Add Attribute Group" button (the bottom button in the list). You will be prompted for a name. An Attribute Group with the name you enter will be added to the current view.

You can use the other buttons to delete the Attribute Group or move it up and down.

Managing Views - You can navigate through your views by clicking on the appropriate tab in the Views area of the UI. To add an empty view to the list, press the "Add View" button. You will be prompted for a name, after which a tab will appear with the name you entered. The name maybe slightly different than the one you typed. View names must start with letters and can only consist of letters or numbers. If you enter an invalid name, the name will be made "View Name" compliant, and then the tab will appear.

The "Copy View" button will duplicate the current view as a new view with the name you provide. This is a quick way to use other views as the basis for others. For example, if you want the Anim and Effects views to be similar, you create your Anim view, delete the empty Effects view, and then "Copy" the Anim view as Effects.

"Delete View" will delete current view from the UI. If you delete, anything unsaved will not be recoverably.

Saving Views - To save the views that are currently in the Template Builder, press the "Create Template" button. A window will pop up letting you know where the template was saved to. This path will also be echoed to the Maya Script Editor History area for easy cut and paste operations.

The template file will be saved to the path listed in the Template Path line at the top of the Template Builder UI. By default, they UI will come up with the standard Maya Program/scripts/ AETemplates directory. By saving them here the templates will be immediately available for use.

If you browse to a different directory to save your templates, you may want to check the "Add to MAYA_CUSTOM_TEMPLATES path" checkbox to ensure that the template will be available immediately saving.

Note that the next time you start Maya the MAYA_CUSTOM_TEMPLATES path will be reset to being empty unless custom scripts are used to set the MAYA_CUSTOM_TEMPLATES path Environment variable.

You may also note that you may need to run a "refreshCustomTemplate" command. If you are authoring a template for a node or node type that has no pre-existing Attribute Editor Attribute Views, there is no need to refresh the template. If, however, a template file exists for a node and a custom view has been used in your current Maya session, you will need to refresh the view in order to see your changes.

To refresh the Template for a node, display a custom view of that node in the Attribute Editor (If the current view is set to the default view a refresh will not occur). From the Maya Script Editor, Command Line or Shelf Button, run the command "refreshCustomTemplate;" and the new template file for that node will be used by the Attribute Editor.

Advanced Templates: Callbacks are a power feature providing a way of injecting custom UI into the Attribute Editor, as well as allowing you to link MEL or Python code to attributes. You can learn more about setting up callbacks in the Maya documentation.

To apply a Callback, highlight the Attribute(s) in the current view and click the Right Mouse button. From the menu that pops up choose the "Add Callback" item. A small window will come up prompting you for the Callback command. If you are creating a python-based callback command, you must include the initial "py." for the callback to work.

After you enter your command, an asterisk (*) will be added to end of the attribute in the list to let you know that a callback exists for this attribute. Use the other Right Mouse Menu items to list callbacks in the current view or in all views.

Advanced Templates: Node Named / Object-Based Views - View templates can be saved for either Node Types, or for uniquely named nodes of a specific type. For example, Transform nodes can have a set of views, while the Transform node named "Supermover" can have a completely unique set of views.

To create Named Node views, click the "Name-Based Template" checkbox underneath the Node Name text field. This will activate the Node Name field. It will default to the name of the node that was selected when you populated the UI, but once it is activated you may enter any name you would like.

Now, when you press the "Create Template" button, a template will be saved that will work for nodes of the current Node Type listed in the UI that are have the name entered in the Node Name text field.

Script Editor: Print Option Vars

This tool will print out the optionVars that are currently being used by any and all tools or UI in Maya. This can be useful when setting up options for custom tools or for investigating the underpinnings of an existing tool.

To use, open the **Print Option Vars** window and either click "All" to print every optionVar, or enter a specific string and click "Search" to print only optionVars containing that word. The results will be printed in the script editor.

Script Editor: Print Global Vars

This tool will print out the MEL global variables currently being used by any and all scripts/procedures in Maya. This can be useful when writing custom tools or for investigating the scripts used to write existing tools.

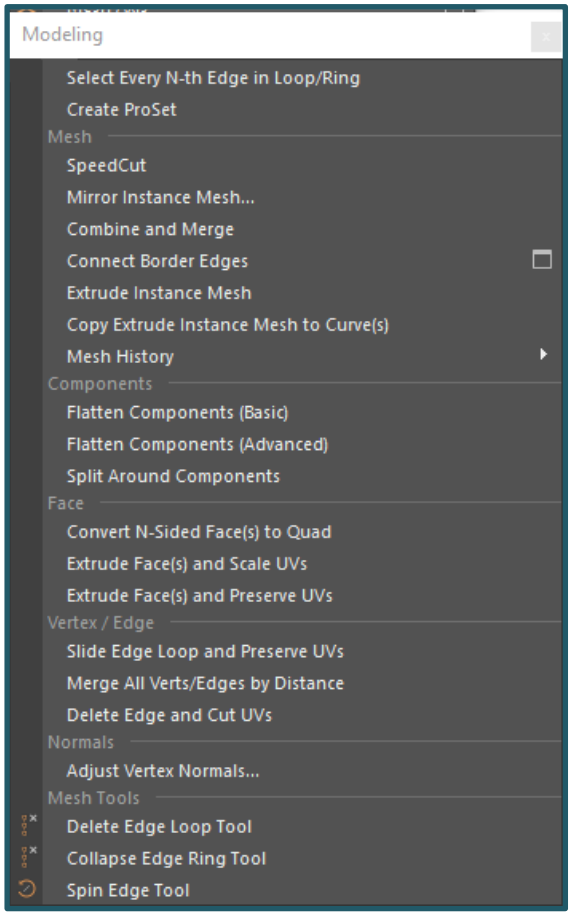
To use, open the **Print Global Vars** window and either click "All" to print every global variable, or enter a specific string and click "Search" to print only global variables containing that word. The results will be printed in the script editor.

Calculator

This opens a simple calculator. The One Liner button allows for an expression to be entered in the field and then calculated.

To use, simply open the Calculator window and either enter numbers with keypad or by clicking.

Modeling



ProSets (New)

ProSets is a new procedural selection set plug-in that facilitates non-destructive, procedural modeling workflows. Interactively adjust multiple selection types with an intuitive drag and drop UI.

Getting Started:

- Create a polygon mesh.
- Add a modelling operation to the mesh (such as bevel or extrude)

Then do one of the following:

Option A

- Create a ProSet node via BonusTools > Modeling > ProSets
- Drag the mesh from the outliner into the drop zone of the ProSet node.
- Select your modelling tool from the pop up window and click continue.

Option B

- Create a ProSet node via BonusTools > Modeling > ProSets
- Drag the modelling tool (eg. polyExtrude) from the Node Editor into the drop zone on the ProSet node.

To test everything worked, click 'Modulus'. Every second face should now be selected.

Notes:

You must manually set the Component Type. i.e. if you've bevelled an Edge you must set Component Type to Edge. There is no way around this. Multiple modelling tools can be added, even if they're on different meshes, but they MUST be of the same component type. Adding multiple modelling tools from the same mesh to the same ProSet can cause graph cycles, these are usually safe, but generate a lot of warnings, so you might want to avoid them.

Check out the **ProSets Getting Started** video on Vimeo

Select Every N-th Edge in Loop/Ring

This tool propagates the current edge ring/loop selection. For example, select an edge, skip the next parallel edge and then shift select the next parallel edge. The result will be that every other edge in the ring is selected.

Update: The "tool" mode is no longer supported because of reliability issues. This now works only as a select and apply operation.

To use, select edges and open **Select Every N Edge**s.

Speed Cut

This feature provides a flexible, comprehensive system for creating and managing boolean objects.

NOTE: This is a Windows only feature

Simply define an existing mesh as your base object. Then use a range of features to create "cutter" objects that will be integrated with the base mesh through boolean operations.

Features in include:

- Base mesh display and selection control
- Base mesh symmetry options
- Cutter object creation and manipulation
- Drag and drop cutter object placement/alignment on base mesh
- Procedural beveling and smoothing for cutter objects
- Cutter alignment controls
- Cutter pattern controls including linear and radial
- Custom cutter creation tools through interactive curve drawing
- Baking of cutter objects to target base mesh

- Retargeting of cutter object from one base mesh to another

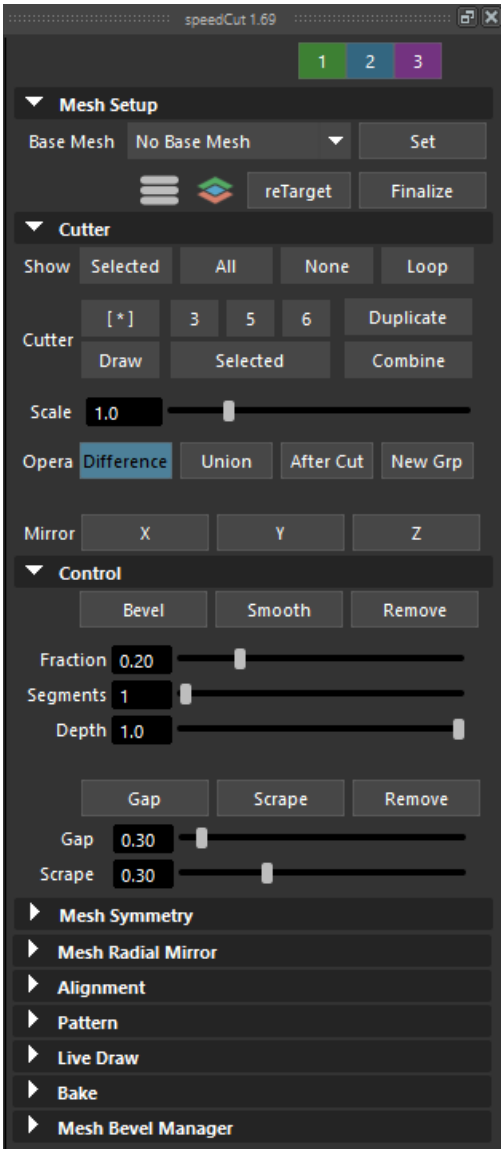
Update: Speed Cut has been updated from version 1.58 to version 1.69 in BonusTools 2025

Fixes / improvements include…

- 1.59 resizeable UI
- 1.60 group color match layer color
- 1.61 fix bug when create New Grp
- 1.62 more bug fix with create New Grp and assign color
- 1.63 fixing bug for prebox not created in baseMesh center
- 1.64 adding rdMirror
- 1.65 improve rdMirror UI
- 1.66 improve dock UI
- 1.67 fix bug dock UI bug in mata 2022+
- 1.68 support add multi meshes to cutter
- 1.69 small bug in python 3

To use, define a base (target) mesh and use Cutter objects to create a variety of Boolean effects.

Note: Check out these youTube video for **SpeedCut tips and tricks**



Mirror Instance Mesh

This tool cuts a mesh in half and makes a mirrored instance for symmetrical modeling. The mesh can be mirrored along any axis in either the positive or negative direction. The result is two meshes, the original cut in half, and a mirrored instance that is linked to the original via constraints for unified transformations. Note that the tool assumes the mesh is at the origin and does all mirroring in world space.

The Combine and Merge tool can be used to rejoin the two mirrored meshes into a single mesh.

To use, select a single mesh and open **Mirror Instance Mesh**.

Combine and Merge

This tool will take two meshes and combine them into one. If they share a series of coincident edges/vertices, then it will also stitch (or merge) the coincident edges/verts together.

The tool can be used for any situation where you want to combine and merge meshes together (e.g. an arm to a torso) but it is particularly useful as a companion tool to Mirror Instance Mesh, in order to merge the meshes back into a single, unbroken piece with no seam.

To use, select two meshes and apply **Merge and Combine**.

Connect Border Edges

This tool creates a new mesh between two selected polygon edge border regions. The borders can be within the same mesh or on two separate meshes. The command will isolate two distinct border regions (internal edges are ignored) and create a new polygon mesh between them.

Auto Reverse: Reverses the ordering of one border line based on the distance of the end points

Reverse: Forces the reverse

Map Edge Count: Uses the edge count of the border parts to determine which edges belong together - instead of just the distance

Quads: Convert the resulting triangles to quads

Quad Angle: The angle for the "Quadrangulate"

To use, select edges to be connected and open **Connect Borders**. Only border edges will be connected so selection does not have to be precise.

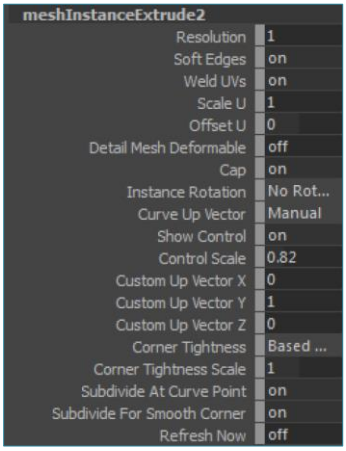
Extrude Instance Mesh

Note: The **meshInstaceExtrude** plugin must first be loaded from the Plugin Manager in order to use this tool. Load the plugin and either restart Maya or use **BonusTools->Help->RefreshBonusToolsMenu** to force a rebuild of the BonusTools menu so that the related menu item will be accessible.

To create an extruded mesh, select an input mesh instance and any number of detail meshes. The input mesh instance must be oriented along the Z axis. Next, select a NURBS curve to use for the extrusion path. The curve can be either a linear curve or a curve of higher degree. Create an extruded mesh by clicking on the Mesh > Extrude Instance Mesh menu item. The input mesh and detail meshes are extruded along the extrusion curve, with the curve passing through the shape's center of rotation.

The texture coordinate space of the profile mesh is replicated along the extrusion, with the coordinates around the profile mesh being repeated along the extrusion and the coordinates along the profile mesh being extended along the extrusion at the same scale as they are defined by the profile mesh. Figure 6 shows a checker texture being applied to both shapes.

To create an extruded mesh with existing setting, select the mesh and click on Mesh > Mirror Extruded Mesh Settings to Curve(s). The upstream network will be duplicated, including the groupParts and groupIds for shading assignment. The upstream input base geometry and input detail mesh will be shared, not duplicated, i.e. only their connections are copied.



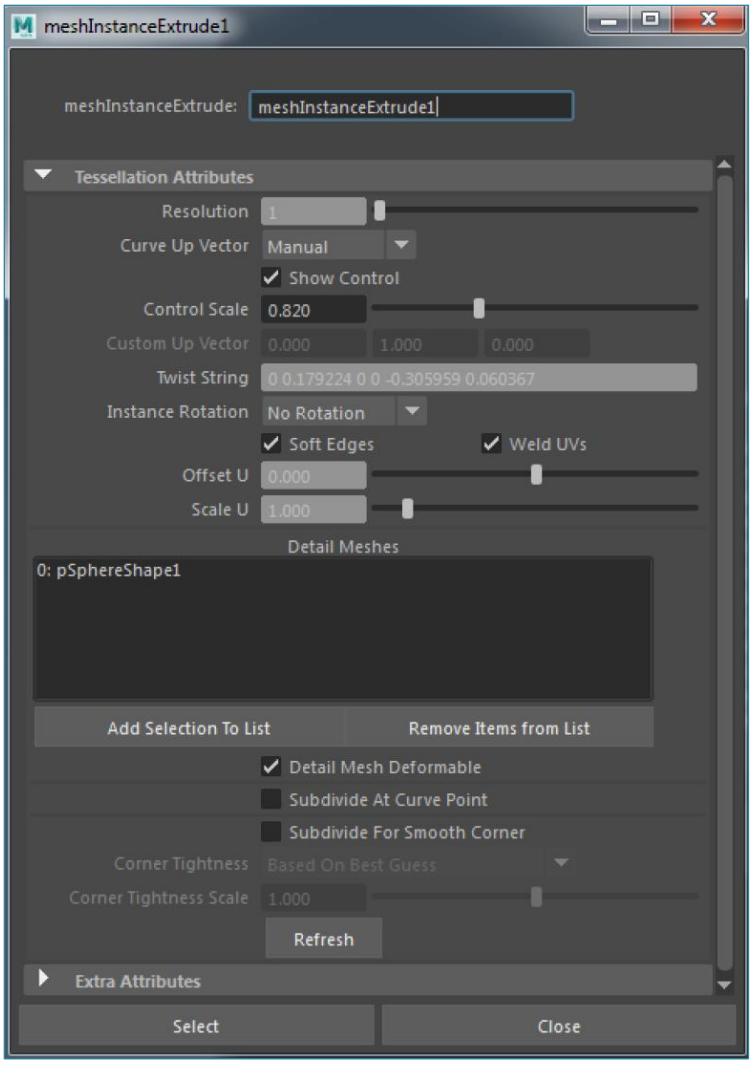


Figure 1. The attribute editor and channel box of the meshInstanceExtrude node

Controls

Resolution – It defines the number of segments between two adjacent edit points of the input curve. The control allows better approximating of the curve curvature. It has no effect on degree 1 curve because degree 1 curve is supposed to be linear and additional segments are of no use.

Curve Up Vector – It defines the initial up vector for the input curve when computing its actual tangent, up vector, and normal vector. The extrusion could fail or provide undesired result if the up vector is too close to the curve tangent, which is the direction of the curve.

Show Control – Show Control is enabled if Curve Up Vector is "Manual". It toggles on and off the circles for controling the up vector at each curve vertex. The circle controls are automatically created on the fly the first time the toggle is on. Each circle is an individual group with the name "circleControl". The circles resets if the curve's topology changes. Each circle is composed of a transform in a group. The transform Y axis control the

twist of the tangent. If the default circle size needs to be changed, it can be achieved by scaling the group node.

Control Scale – This is the scale factor for all the connected circle controls.

Custom Up Vector – The vector to be used for up vector if Curve Up Vector is "Custom".

Twist String – The string lists all the twist angles. It should be automatically updated when the circleControl is updating the twist angle. If the string is updated, it should also automatically update the twist angles. This string attribute allows to transfer the existing twist angle setting from one extruded geometry to another extruded geometry.

Instance Rotation – The rotation of the instance mesh and detail meshes before extrusion. Instance Rotation lets the user to rotate these shapes in 90 degree increments as it is extruded along the curve.

Soft Edges – If it's on, the normal at the segment boundary will be interpolated to give smooth shading.

Weld UVs - It specifies whether the texture coordinates will squash and stretch to match the curvature. If Weld UV is on, the uv at segment border will be continuous.

Offset U – the offset in texture space along the curve.

Scale U – the scale in texture space along the curve.

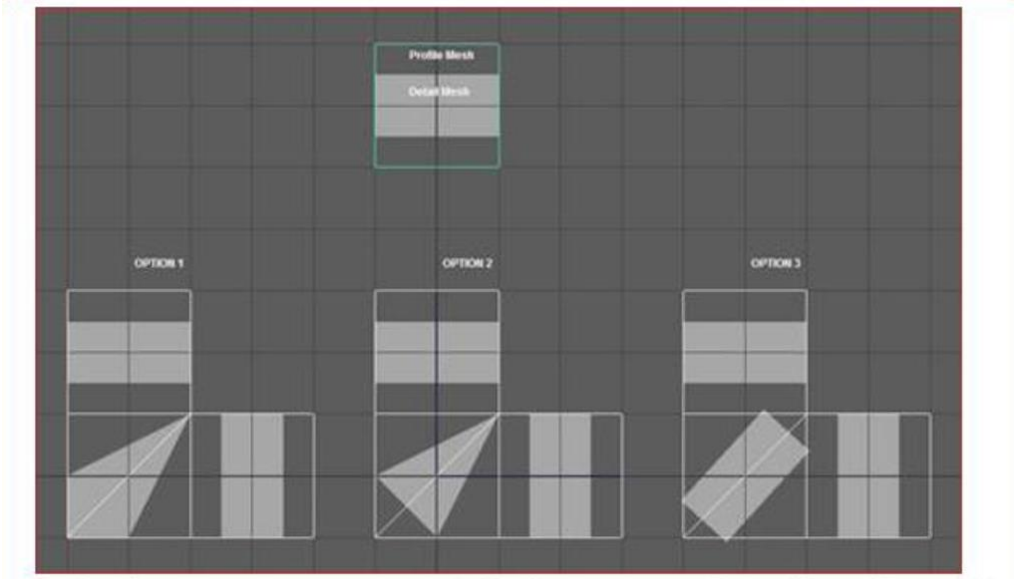
Detail Mesh list – The list shows the detail mesh(s) associated with the extruded mesh.

Add Selection To List button – Click on the button to add all the selected mesh(s) in the active selection list as detail mesh(s).

Remove Items from List button – Click on the button to remove all the selected mesh(s) in the Detail Mesh List as detail mesh.

Detail Mesh Deformable – The detail mesh when extruded has the option to deform the same way as the input mesh instance or be treated as rigid body. See Figure 3.

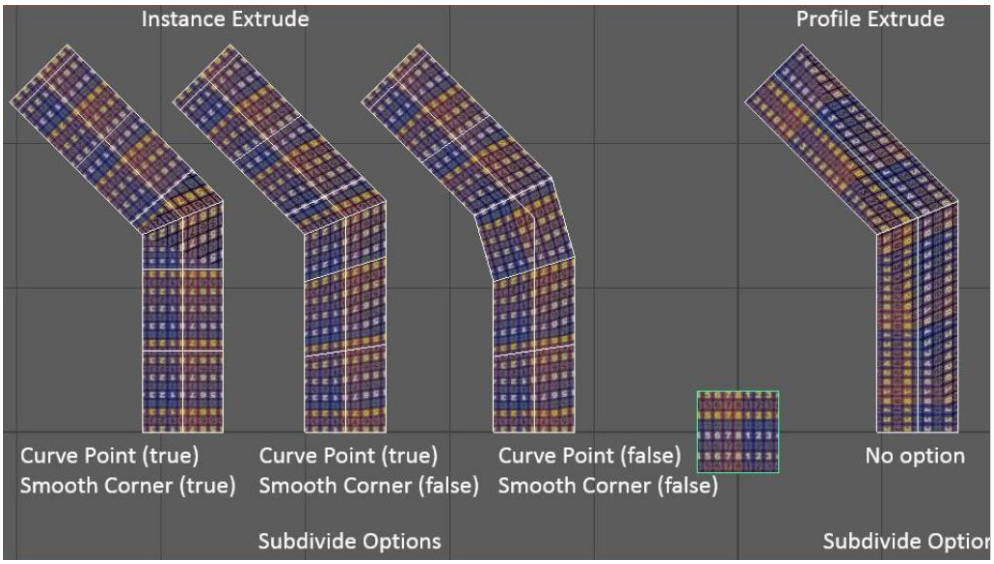
Subdivide At Curve Point – This decides if the subdivision should be applied to curve point.



Subdivide For Smooth Corner – This decides if additional care should be applied to achieve smooth corner. See Figure 4.

Corner Tightness – The inside and the outside of a tube bends at different rates at a corner depending on the "radius" of the tube. Corner tightness decides what the radius should be based on. It could be the width, height, or diagonal size of the input mesh. The best guess option will choose among these three.

Corner tightness Scale – It's a multiplier applied on top of the "radius" from corner tightness.



Refresh – This button will reconstruct the extruded mesh. This should be needed only in rare occasions. For example the shading assignment changes in the input mesh.

Refresh Now – This control is only available in channel box. It's off by default. If it's set to on, it triggers the rebuild of the associated extruded mesh and once it's done, the control is set to off again.

Hidden attributes

Cap – It decides if the extruded mesh should contain the cap or not. The same effect can be achieved by deleting the caps in the input mesh.

Input up vector curve – The input curve to manually control up vector. The input up vector curve must have the identical topology as the input curve. The corresponding direction defined by the vertex pair, i.e. going from the nth vertex at the input curve to the nth vertex at input up vector curve, is used as the up vector at the vertex for the extruded mesh.

Mesh History:

Delete Upstream Mesh History

This tool deletes upstream history of the selected node in the channel box. Anything before the selected node will get "baked" and the selected node as well as anything downstream will be preserved. This allows for more control of deleting history.

To use, select a mesh and an input node in the Channel Box and apply **Delete Upstream Mesh History**.

Insert Intermediate Mesh Shape

This tool inserts a poly shape node upstream of the selected node in the channel box. This allows for modifications to the middle of history chains.

To use, select a mesh and an input node in the Channel Box and apply **Insert Intermediate Mesh Shape**

Copy/Paste Mesh History

Copy/Paste Mesh History is actually two separate commands that will copy and paste polygon history between meshes. These were originally intended for editing **blendshapes**, but can be used with regular poly objects. In the past, if the base mesh or a target mesh for a blendshape, required a topology change, this tool could be used to apply those changes to every mesh in the blendshape. This can now be done directly in Maya with BlendShapes->BakeTopologyToTargets.

This tool can still be used for copying basic poly edits from one mesh to another. To use, first select the mesh with the history you would like to copy. Graph this mesh in the hyperGraph. Select the history nodes in order (first to last) and apply **Copy Mesh History Nodes** to Copy. Second, select the meshes to copy the history to and apply **Paste Mesh History Nodes**.

Notes:

- For the history to be properly copied, vertex tweaks need to be recorded as nodes. To do this, use Edit Polygons->Move Component.

- These commands assume that mesh topologies are basically the same. This means that they have the same number of vertices with the same vertex order. You will have varying degrees of success if the topologies differ.

Flatten Components - Basic

This tool can be used to quickly flatten a selection of polygon components along a calculated plane. The virtual plane will be at the center of the selected components and oriented based on an average normal. This tool can be used on faces, edges or verts and can be repeated on the same object.

Update: The undo issue have been fixed in 2025. You can now undo after flattening components

To use, select 2 or more poly components from a single object and apply **Flatten Components (Simple)**.

Flatten Components - Advanced

This tool can be used to flatten a selection of polygon components along a virtual plane. The plane will be created automatically at the center of the components and oriented based on an average normal. A manipulator will be created so that you can then interactively adjust the position and orientation. This tool can be used on faces, edges or verts and can be applied more than once to the same object. The snapping of the verts to the plane is done with a TransferAttributes node and this node will will stay "live" and in order to remove/delete the control plane and bake in the flatten effect you will first need to delete history on the mesh.

Update: Improved usability in 2025. The resulting flatten plane is now automatically selected and set to move the flattened components along the average normal

To use, select 2 or more poly components from a single object and apply **Flatten Components (Deformer)**.

Split Around Components

This tool finds the edge ring around the selected components and then split it.

To use, select some polygon components and apply **Split Around Selection**.

Convert N-Sided Face(s) to Quad

This allows you to easily convert any n-sided face to quads. Select an mesh or face(s) and it will automatically isolate any N-Sides faces and then apply a Triangulate followed by a Quadrangulate in succession.

To use, select a mesh or face(s) and apply **Convert N-Sided Faces to Quad**.

Extrude Face and Scale UVs

This tool extrudes faces and scales the resultant UVs so that the new UVs are not overlapping.

To use, select face(s) to extrude and apply **Extrude Face and Scale UVs**.

Note: There may be unintended scaling issues when applying to faces at the edge of a texture border. This is a known limitation of the tool.

Extrude Face and Preserve UVs

This tool extrudes faces and but first creates a preserveUV node (via Transfer Attributes) so that the associated texture / UVs will not be inadvertently effected by the extrude node. The underlying UVs will be corrected based on changes to the faces that are extruded. This particularly noticeable with offset and scaling of the extrude manip.

To use, select face(s) to extrude and apply **Extrude Face and Preserve UVs**.

Slide Edge Loop and Preserve UVs

This tool slides the selected edge loop along it's connected edges but first creates a preserveUV node (via Transfer Attributes) so that the associated texture / UVs will not be inadvertently effected by the slide. The underlying UVs will be corrected based on changes to the edges.

To use, select face(s) to extrude and apply **Slide Edge Loop and Preserve UVs**.

Merge All Verts/Edges by Distance

This tool merges all of the verts on a mesh that are closer than the selected 2 verts or 1 edge. The distance between the selected verts is used as the merge tolerance in determining what other verts on the mesh should be merges.

This tool has been updated to work with an edge as well.

To use, select two verts or a single edge and apply Merge Vertices by Distance.

Delete Edge and Cut UVs

This tool deletes an edge but first cuts the bordering UVs. This is useful when deleting a border edge.

To use, select an edge apply **Delete Edge and Cut UVs**.

Adjust Vertex Normals

This tool creates a window for interactive vertex normal adjustment.

To use, open **Adjust Vertex Normals**. Select a vertex or a face vertex. The sliders will update to show the current normal values. If more than one item is selected, only the values of the first will be displayed. Changing the values will change the values for all selected.

Delete Edge Loop Tool

This tool allows you to easily delete all edges associated with the loop of a given edge. When the tool is activated you can click on any edge and the tool will automatically convert it to the loop and then delete the associated edges. The tool will remain active so you can continue to delete additional edges loops.

To use, start the **Delete Edge Loop Tool**. When the tool is active, simply click on an edge and its associated loop will be deleted.

Collapse Edge Ring Tool

This tool allows you to easily delete all edges associated with the ring of a given edge. When the tool is activated you can click on any edge and the tool will automatically convert it to the ring and then collapsed the associated edges. The tool will remain active so you can continue to collapse additional rings.

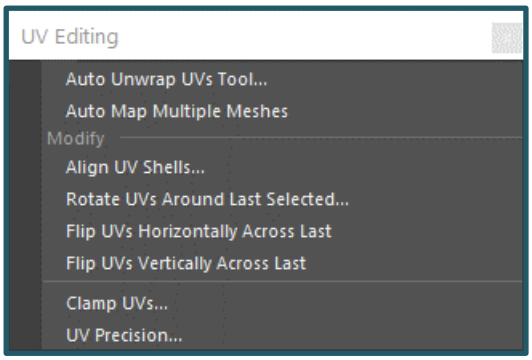
To use, start the **Collapse Edge Ring Tool**. When the tool is active, simply click on an edge and its associated loop will be deleted.

Spin Edges Tool

This is the tool version of the spin edges command. When the tool is active simple click on an edges and it will "spin" so that it now connects to the next vertex in each direction. The tool will remain active so you can continue to spin additional edges (or simply hit the g key to continue the spinning the same edge).

To use, start the **Spin Edges Tool**. When the tool is active you will be prompted to select faces and edges for wedging.

UV Editing



Auto Unwrap UVs Tool

This tool can be used to quickly and easily create UVs shells for a variety of objects including roads, pipes, horns, trees, bodies and heads. Shells can be automatically unwrapped in various ways based on a number of different options. The tool will guide you through the steps of unfolding complex objects and will keep track of the steps along the way, thus allowing you to step back and try alternative layouts. The UV Shell Creation method will determine how the resulting UVs get created.

Update: A bug that was preventing the Uniform and Legacy methods from working has been resolved along with a few additional minor bugs fixes.

Also fixed - //error line 950: This command requires at least 1 argument(s) to be specified or selected; found 0. //

UV Shell Creation

- **Unfold 3D:** Uses the Unfold 3D algorithm that was introduced in Maya 2015 to spread out the resulting UV shell. This method is good for many types of objects including heads, torsos, horns, tree branches, arms or legs.
- **Legacy Unfold** : Uses the legacy unfold algorithm to spread out the resulting UV shell
- **Legacy Unfold with Symmetry Anchors** : Uses the legacy method but also takes symmetry into account by initially aligning along a specific axis (e.g. a face or head.) This option also adds the additional step of defining points of symmetry (e.g. tip of the nose or center of the chest)
- **Uniform Faces**: Tries to make each face in the UV shell the same size in UV space. It can be used on closed or open meshes. This method is good for things like uniform roads, walls or pipes
- **Pipe / Tube Contour** : Tries to make faces proportional in the UV shell to match that of the 3d space while at the same time maintaining the contour or flow of the pipe in UV space. This method is good for things like bending pipes, tubes, hoses, coils, etc… This method uses the new Contour Stretch algorithm from Maya under the hood. It requires a mesh or faces forming a tube or pipe shape with 2 open ends/holes, along with the selection of a single edge loop that runs along the length of the pipe/tube shape. Note that a limitation of this option is that it can only be used to unwrap one pipe or pipe section at a time resulting in a single UV shell at a time. It can however be repeated for multiple sections of a pipe or tube.

Initial Projection (Note: Unfold 3D automatically handles symmetry without requiring this extra step)

- **Facing X** (used only by advanced unfold mode)
- **Facing Y** (used only by advanced unfold mode)
- **Facing Z** (used only by advanced unfold mode)

UV Shell Layout

- **Proportional Fit** : Layout new UV shells along with any other existing shells for that object based on actual 3D world space size
- **Uniform Fit** : Layout new UV shells uniformly (no distortion) in UV space between 0-1
- **Stretch Fit** : Layout new UV shells non-proportionally (stretch to fit) in UV space between 0-1
- **None** : Do not alter size or layout shells at all in UV space

Display Settings

- **Set Edge Color to Red** : Temporarily sets selected edge color to red for better visual feedback while tool is active
- **Set Mesh Shader to Green** : Creates preview shader for better viewing of border edges
- **Create Checker Shader** : Creates check shader for better viewing of resulting UVs
- **Isolate Select Object/Faces** : Isolates only selected object or faces while tool is active and hides all others

Auto Map Multiple Meshes

This tool creates an Automatic Mapping node to control the mapping on multiple meshes.

In older versions this tool did not work as advertised and gave very unpredictable results. It has been completely rewritten in this release to achieve the same goal but using a different method. It can be used to create a single UV set with no overlapping UVs for multiple meshes. The most common and practical use for this would be creating secondary UVs for light maps that span across several objects. The resulting UVs will be controlled with a single auto UV projection node that will be applied to a temporary object called **autoProjectautoMapMesh** and then transferred to the original meshes via transfer attributes.

Important: In order to "bake" in the UVs you will need to delete history on the target objects.

To use, select meshes and apply Auto Map Multiple Meshes.

Align UV Shells

This tool aligns a UV Shell to the left, right, top and bottom boundary of another UV shell.

Update: The UI layout has been improved to be more intuitive. This tool now works with any component selection, faces, edges, verts or UVs. You can also now align several shells at the same time.

To use, open the Align UV Shells window... Select one or more components on different shells in the Texture Editor and select the alignment option from the window.

Note: This is a python based tool and may not work after default installation. See installation instructions on setting up your python path if you want to use this tool.

Rotate UVs Around Last Selected

This item will rotate UV coordinates in the UV Texture Editor using the last selected UV as a pivot.

Note: This tool requires that the global selection pref called "Track Selection Order" be turned on. This is off by default so the tool will first check the preference and give you the option to enable it before using the tool. This should only need to be done once.

To use, open **Rotate UVs Around Last Selected** window. In the UV Texture Editor, select the UVs to rotate and select a UV to act as the pivot point as the last selection. Enter the rotation value in the window and press Rotate UVs.

Flip UVs Horizontally Across Last

This item will flip UV coordinates horizontally in the UV Texture Editor using the last selected UV as a pivot.

Note: This tool requires that the global selection pref called "Track Selection Order" be turned on. This is off by default so the tool will first check the preference and give you the option to enable it before using the tool. This should only need to be done once.

To use, select UVs make desired UV pivot the last selection, then apply **Flip UVs Horizontally Across Last**

Flip UVs Vertically Across Last

This item will rotate UV coordinates vertically in the UV Texture Editor using the last selected UV as a pivot.

Note: This tool requires that the global selection pref called "Track Selection Order" be turned on. This is off by default so the tool will first check the preference and give you the option to enable it before using the tool. This should only need to be done once.

To use, select UVs make desired UV pivot the last selection, then apply **Flip UVs Vertically Across Last**

Clamp UVs

This item will clamp the selected UVs to a user specified bounding box.

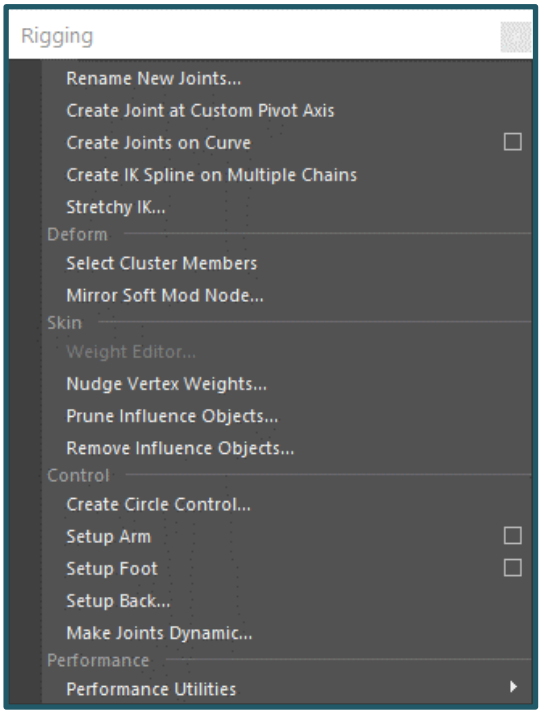
To use, select the UVs to clamp and open **Clamp UVs...** Enter the bounding box coordinates (U min., V min., U max., V max.) and press OK.

UV Precision

This item will adjust the precision of selected UVs.

To use, select the UVs to change and open **UV Precision...** Enter the number of decimal places that should be kept for the UV coordinates and press OK.

Rigging



Rename New Joints

This will create a window to allow for automatic renaming of the selected object. This was created to quickly and easily rename joints in a skeleton. When creating a spine, the user general wants the joints named spine instead of joint. It's also good for fingers.

This does not have to be limited to joints. Any item that is selected will be renamed so it could be useful for level building activities.

To use, open **Rename New Joints**. Enter a name to rename the joints to and start placing joints (or objects). The window provides an on/off toggle at the top so that the renaming can be easily interrupted.

Create Joint at Custom Pivot Axis

This too creates a new joint at the current active pivot. If a custom pivot has been set the locator will inherit the custom orientation and position of that pivot. This can be useful when rigging hard surface meshes with specific angles / orientations, by quickly creating a joint to match the position and orientation of a given component. Credit: this is based on the script create_pivot_bone created by Randall Hess

To use select a mesh component or object, customize the pivot if desired, then click **Joint at Pivot Axis.**

Create Joints on Curve

Formerly called "Bones on Curve" this tool will place a user specified number of joints/bones on a curve with the option to rebuild the curve and add spline IK to the joints. When rebuilding the curve, the degree is changed to 2.

To use, select a curve, choose number of bones in **Create Joints on Curve** window, then apply.

Create IK Spline on Multiple Chains

This tool can be used to create multiple IK Splines for several joint chains at once. This can be useful for things like hair, antenna, grass, branches, etc… where you are dealing with a lot of joint chains.

Note: Unlike the standard IK Spline tool you only have to select the root joint. The end joint will be found automatically.

To use, select the root joint of the joint chains you wish to use and apply **Create IK Spline on Multiple Chains**

Stretchy IK

This tool makes the selected IK handle's chain stretchy. It requires the user to specify the axis that points down the bones in the chain.

To use, select an IK handle, choose axis and apply Stretchy IK.

Select Cluster Members

This tool will select the components of the active cluster.

To use, select a cluster and apply **Select Cluster Members**.

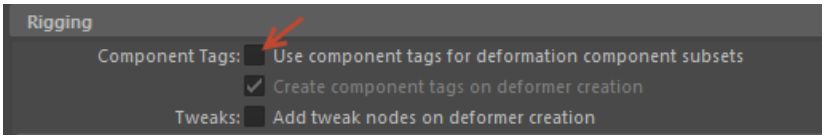
Mirror Soft Mod Node

This tool will duplicate and mirror an existing soft mod node across a mesh in either object space or world space and along any axis. This allows you to symmetrically apply a soft modification node to something like a character's face or body. Note: This is meant to work with a Soft Mod –**Node**- not a soft select.

To use, select an existing Soft Modification Node and open the **Mirror Soft Mod Node** window.

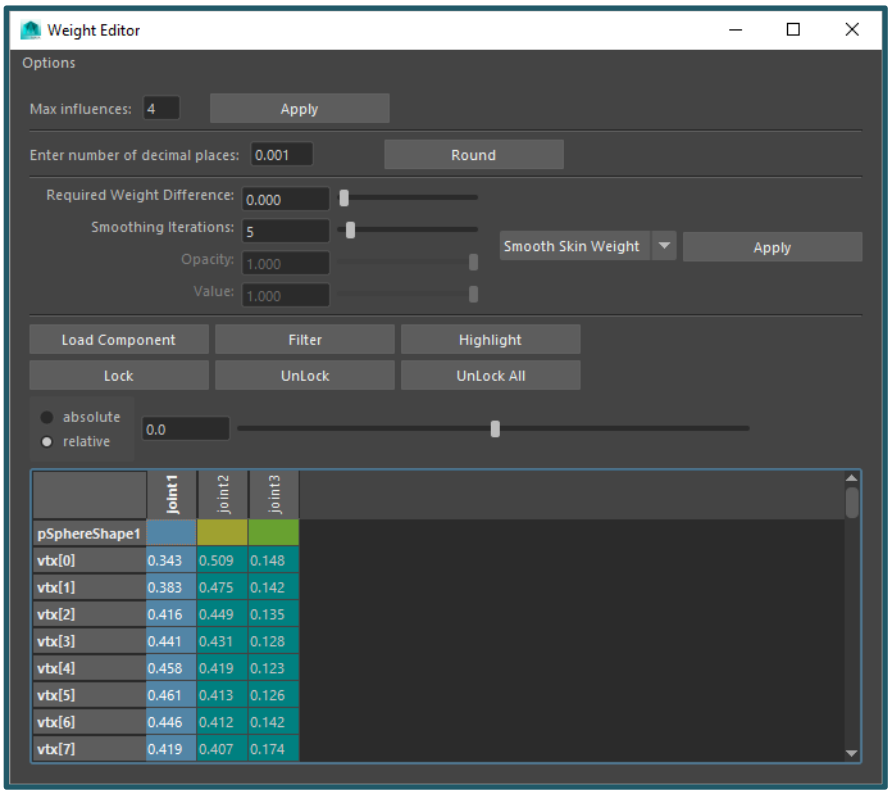
Weight Editor

Note: This plugin does not work with skin clusters that use the new **Component Tags** for deformer membership. It will continue to work if skinning is set to legacy method. This can be enabled/disabled in the Maya preferences under rigging. The Component Tags setting needs to be off before skin binding.



This tool will allow you to see and adjust all of the weights on a skinned mesh. It is intended to be an alternative to the component editor and easier way to view and modify weighting values. You can do a number of different adjustments to your weighting including:

- Viewing Weights
- Filtering Weights
- Adjusting Weight Values
- Setting Max Influences
- Setting weight precision
- Smoothing Weights
- Locking / Unlocking Weights
- Highlighting Joints



To use, select a mesh that has been skinned and open the **Weight Editor** Window.

Nudge Vertex Weights

This tool opens a window to adjust vertex weights for a selected influence object by a user defined amount.

Update – this tool was mysteriously missing in previous versions. It is now working in 2025. The UI has also been updated to be more intuitive and the script now checks to make sure you have the correct selection types (e.g. vertices and joints)

To use, select a vert and an influence object (joint) then open the **Nudge Vert Weights** Window. Enter a nudge value and increase or decrease weight for selected influence object.

Prune Influence Objects

This item will allow the user to set smooth skin weighting to 0 based on a number of influence objects. This is an alternative to Skin->Edit Smooth Skin->Prune Small Weights which does not provide a mechanism for pruning to a number of influence objects.

Prune below is included as a convenience and works the same as in the Prune Small Weights tool.

To use, select a skinned mesh and open the **Prune Influence Objects** window. Maximum Influences is the number of joints (or influence objects) affecting the skin. Press Prune Influences to reweight the skin.

Note that this script is not undoable - undo is turned off to improve performance which can be slow with this command.

Remove Influence Objects

This item allows for selective removal of influence objects from a smooth skin. Use this for a multi mesh character to remove specific influence objects. When you smooth skin multiple meshes, every mesh is attached to every bone in the skeleton. This can slow performance and complicate skin editing.

This is similar to Skin->Edit Smooth Skin->Remove Unused Influence Objects but it will let you selectively remove influence objects. If the character's head is a separate piece of geometry, the feet should not deform it so they can be removed as influence objects. The neck or shoulder joints may not be initially affecting the head mesh but they may be required to so they may be retained initially as influence objects.

To use, select the skinned geometry and open **Remove Influence Objects**. A window will open. With the geometry selected, press Load New Skin. The window will show all the influence objects for the selected geometry. Highlight the influence to remove and press Remove Influence.

Only one influence can be removed at a time.

Create Circle Control

This tool will create a nurbs circle for each selected transform/joint that will act as a control for the target object. Each circle will be tagged with a "control" node. See docs for control node for explanation and use cases.

Shape Under Transform: You can choose to parent the shape of the circle under the target transform/joint. This will allow you to directly select the joint transform by selecting the corresponding circle shape.

Parent Constraint: You can drive the target transform/join with a parent constraint. If this method is chosen a secondary group/hierarchy will be created to mimic the hierarchy of the joint chain. This is only applicable when working with groups of joints or transforms. If a hierarchy exists the "control" nodes will be connected so that you can pick walk easily between them without selected the dummy group nodes. See docs for control node for explanation and use cases.

Circle Size/Color: Set the desired size and color of the resulting nurbs circle

Circle Orientation: Determines how the circle will align/orient with the target joint / transform

Setup Arm

This tool can be used to automatically set up an arm rig.

To use, select the shoulder joint, followed by the wrist joint and apply **Setup Arm** with the settings below. The tool will add the appropriate controls. When finished, the Prefix Hierarchy option box will open so that the nodes may be named.

Twist Joint: This specifies whether there is a twist joint in the fore arm.

Hand Control: A user specified hand control node may be created. Place it in the desired position relative to the skeleton. If left blank, a selection handle will be used.

Elbow Control: A user specified elbow control node may be created. Place it in the desired position relative to the skeleton. If left blank, a locator will be used.

Note: This assumes that the leg is built in the side window with the with the character facing the +Z axis.

Setup Foot

This tool can be used to automatically set up a leg and foot rig

To use, select the hip joint, followed by the ankle joint and apply **Setup Foot** with the settings below. The tool will select the ankle, ball and toe and add the appropriate controls. When finished, the Prefix Hierarchy option box will open so that the nodes may be named.

Heel Joint: This specifies whether the leg hierarchy has a heel joint between the ankle and ball joint.

Foot Style:

> **IK Handles:** This sets up the foot with three IK handles and a control hierarchy.
>
> **Parent Constraint:** This makes use of the new parent constraint. This allows for easy control between IK and FK. It uses only one IK chain so that the Move IK to FK command works smoothly.
>
> **Set Driven Key:** This foot uses a number of Set Driven Keys so that the user may roll the foot with one attribute.

Each of these options adds control attributes to the foot control node.

Foot Control: A user specified foot control node may be created. Place it in the desired position relative to the skeleton. If left blank, a selection handle will be used.

Knee Control: A user specified knee control node may be created. Place it in the desired position relative to the skeleton. If left blank, a locator will be used.

Note: This assumes that the leg is built in the side window with the foot pointing in +Z axis. The script has some comments on how to change the setup in terms of the foot node and the poleVector locator.

Setup Back

This tool sets up control objects for a character's back, including the advanced twist controls in Maya version 6 that allow twisting controls from both ends of a spline ik.
This information is intended to help you work with or customize the back setup bonus tool. It explains what the script does, what its assumptions are and other ways you can tweak the setup.

To use, you need to have a skeleton already made with a root joint and spine joints. Select the first spine joint next to the root and the last spine joint. Select the options you want and press the Rig Back button. If you are using your own control objects, you must load them into the option box before selecting the joints and pressing the Rig Back button. Once the rig is created you can animate the control objects (default poly cubes or custom objects) or clusters to control your back motion. This tool integrates well with the foot setup tool in bonus tools.

Assumptions and Limitations:

The script assumes that your character has a vertical spine and faces forwards on the +Z axis. The script will flip the involved joints axes so that Y points in -Z and X points down the bone. These orientations are the most common in rigging. No choice is given here, so beware if your direction of your axes differs. It does not work well with bound surfaces as they will twist if the joint axis is reset by the script. You may be able to compensate for this by rotating the joint axis after the fact, but better to create the setup before binding or have the joint axes point the same way as the script would. The script isn't made to redo an existing setup, so you need to either undo, or delete the control objects, clusters, spline IK handle and curve. You can keep custom control objects, but you will need to delete the constraints from the root if you chose "Use Bottom for Root" in the options.

What the tool actually builds:

It creates a curve and the spline IK handle between the selected joints and then creates a cluster for each CV on the curve. It builds polygon cubes for controls and uses them for the start and end controls of the spline IK, or uses objects you load in the options window. It takes advantage of the advanced twist controls for spline IK, allowing twist controls at either end of the spline IK, and adjusts the settings best for a spine. You can see the control objects assigned to the IK handle in the Attribute Editor under IK Solver> Advanced Twist Controls. There are many choices in how you can use this feature, but the script settings should be useful for a spine. The clusters are parented to the top or bottom spine controls. Moving a control gives a gross change in the shape of the spine. The clusters can be used for finer control. Use Bottom Control Object for Root allows you to use the bottom control object as a point and orient constraint for the root/pelvis.

More on using the tool:

To use your own objects as the top and bottom control for the spine, you need to turn on the flag and then select the appropriate object and hit the Get button. Alternatively, you can type in the name of the object and hit Enter. The script won't move your objects, so place them where you want them to be first. Fewer clusters may mean less detail in spine shape, but it's also less to animate and tends to a smoother looking spine. Once you hit the Rig Back button and Maya creates the rig, the Advanced Twist Controls (Attribute Editor for spline IK handle, under IK Solver Attributes) shows the objects listed as the World Up Objects. These are effectively aim constraints for the spline IK. You can adjust the distribution of twisting down the joints with the Twist Value Type. You can use a ramp to distribute the twisting down the joint chain and animate it if you want to change the distribution without moving shoulders and pelvis. If you aren't creating a biped spine you might want to experiment with the various type of twist controls offered.

You can tweak the script or adjust the advanced twist controls after using the tool to modify the controls. Start from the default settings. For instance, on a quadruped you probably want the joint Y axes pointing in world +Y, and pointing close to -Z (default) will cause flipping if you move the top control in Y. To make the handle compatible for a horizontal spine as in a quadruped, change the Up Vectors for the Advanced Twist Controls to 0,1,0 and 0,1,0. This will stop the flipping and allow you control over a horizontal spine. You might find that the setup can be good for the long neck of an animal if you skip using the bottom control for the root.

You may not want to move the whole back with the cube control objects but rather have a control for the whole torso which would be a parent for the back controls. You could re-parent the clusters to a torso control and animate the clusters separately leaving only the first and last cluster parented to the control objects. Or you could animate them while still parented to the control objects. These choices really get down to how you want to animate. If you do use a torso control, you'll need to use the bottom control as a constraint for the root (Use Bottom Control Object for Root in options, or constrain it yourself). You need to remember too, that the back controls are only a part of a character rig and may need to be constrained or parented to higher level controls to stay with the character as it moves.

Make Joints Dynamic

Make Joints Dynamic: This tool can be used to apply secondary dynamic animation to a joint chain or a series of joint chains. The chains can in turn be attached to an animated character or some other moving object, and will react to the existing animation. This can be useful for creating secondary animation for various character elements such as: ponytails, antennas, ropes, scarves, chains, capes... You may get varying results depending on the type of method you choose, so you may need to experiment with each type as well as it's associated settings and attributes. This tool can also be used as a way to use Maya's various dynamic systems for game characters. The results of any of these effects can baked down to the joints themselves via the Bake Simulation tool, and can then be exported to a game engine as simple curves on joint rotation and/or translation.

Note: A attribute exists for hair driven joint chains that allows you to quickly toggle between Nucleus and the Classic hair solver. It was reported that the behavior for Nucleus is different and in some cases not as predictable as the behavior with the Classic hair solver. Now you have the option of switching between the two with a simple pulldown menu in the ChannelBox.

Single Chains:

Each of the single chain types uses a curve that will in turn drive an IKSplinehandle connected to the joint chain. The primary control attributes will then be added to the handle providing easy access for editing the dynamic animation. In each case, the IKHandle, curve, and all associated dynamic objects will be created and connected automatically.

- **Hair** : HairSystem -> Folicle -> Curve -> IKSpline -> Joints
- **Jiggle** : JiggleDeformer -> Curve -> IKSpline -> Joints

Before applying **Make Joints Dynamic** for single chains, you must first select the root of the joint chain that you wish to make dynamic, followed by the end joint. After you choose/ the animation driver type that you'd like to use, you will also have an option for the detail of the curve that will drive the chain (Low = 4 cvs : Med = 5 cvs : High = 6 cvs) Lastly you may set the falloff of the given effect. This refers to the amount that the effect will decay from the end of the chain to the root. The end of the chain will have the most movement, the root will have no movement, and the amount of movement for the joints in between will be determined by the falloff.

Note: In order to modify the falloff afterwards you will need to use the component editor jiggle deformers.

Hair extras: In the case of hair falloff you can simply edit the stiffness ramp widget in the hair system. For hair you can later add collision constraints. Simple sphere stand-ins are recommended whenever possible for better results.

Multi Chains:

The joint chains that you drive with nCloth must be siblings to one another. It is also generally recommended that they have the same number of joints per chain. Before applying MakeJointsDynamic for multi chains, you must first the root of each chain. The order is important as it determines how the nCloth object gets built. The verts of the resulting nCloth mesh will match the joint positions exactly.

The falloff setting determines how much the static input mesh (e.g. original joint positions) will influence, or limit the motion of the nCloth.

There are 3 methods for linking the nCloth to the joints.

- **Spline IK** - this is the less effective method but may be a bit faster than Standard IK. An IK Spline will be created for each chain. The end result will be joint rotations.
- **Per-Joint IK** - this is will create an IK handle for -each- joint/bone in each chain. The end result will be joint rotations.

- **Stretchy Joints** - this will create a locator/constraint for each joint in each chain. The end result will be joint translations so that squash and stretch can be achieved. Note: when baking joint rotations will not be affected.

Once you've created the nCloth you may be prompted to select 1 or more objects for the nCloth to collide with. Simply select the object(s) and click the Make Collide button. To bypass this just click No Collisions. You can always add static mesh collision objects later using traditional nCloth methods.

Performance Utilities:

This set of utilities is related to various animation/rigging performance features in Maya.

Toggle Parallel Evaluation

Toggle the global pref between parallel and DG evaluation. To be used in conjunction with the Frame Rate and Evaluation HUDs to compare performance of animation in scene.

Toggle GPU Acceleration

Toggle the global pref for GPU Overide on/off. To be used in conjunction with the Frame Rate and Evaluation HUDs to compare performance of animation in scene.

Toggle OpenSubdiv + OpenCL for Selected

Toggle the per mesh attributes for OpenSubdiv and OpenCL for each selected mesh to allow for GPU acceleration of smoothed meshes. This will also check the GPU Overide settings and refresh it if necessary.

Profile One Frame

Open the Performance Profiler window and evaluate only a single (current) frame all in one step.

GPU Acceleration Status: Currently Supported

List only the deformer chains that are currently supported for GPU acceleration in the current scene. See results in Script Editor output.

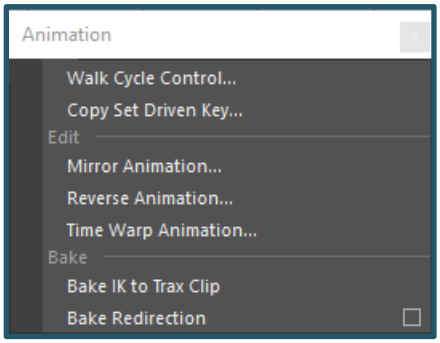
GPU Acceleration Status: Check Selected

Check all selected objects for related deformers and return the GPU acceleration status. See results in Script Editor output.

GPU Acceleration Status: Check All Deformers

Check all common deformers in the scene including skinCluster, blendShape, softMod and cluster nodes, and return the GPU acceleration status. See results in Script Editor output.

Animation



Walk Cycle Control

This is a window to set up character controls and then key them for walk cycles. To be clear, these walk cycles are as if the character is on a treadmill. The character should be symmetrical across the X axis.

When a key is set for one side, a key is set for the opposite side at the opposite keyframe.

- a 24 frame cycle
- frame 5 is active
- user keys left foot (from the control window), left foot is actually the left foot and the left knee
- control window keys the right foot in the same position at frame 17 (current frame + cycle offset)

Note: The order of controls in the control window is important as it determines the symmetrical relationships (i.e. mapping)

- Left Foot set to Left Foot and Left Knee
- Right Foot set to Right Knee and Right Foot
- This will be a problem as the Left Foot is being mapped to the Right Knee

Copy Set Driven Key

Copies set driven keys from one channel to another.

Set Driver Key Source: This is the channel to copy the Set Driven Key from.
New Destination Driver: This is the new driver for the target channel (New Destination Driven).
New Destination Driver: This is the channel to copy the Set Driven Key to.

To use, open **Copy Set Driven Key**.... then load the required objects and press Copy.

Mirror Animation

This tool allows you to take an existing animation and mirror it along a given axis. For example, if you have a character that is walking from left to right, and you want the identical walk from right to left. This script can be

used to invert the animation. This script addresses two different situations where you would want to mirror/invert animation on a character.

- **Between:** Swapping animation from the left side to the right side or vice versa (eg... IK handles, constraints for hands and feet, shoulder and elbow joints). This requires swapping all anim curves from side to side, as well as inverting certain curves based on the axis you want to mirror across. In this case you would select one pair at a time and then apply the mirror. For an entire character you would need to select each pair that required mirroring (eg... hands, elbows, feet, knees, etc...)
- **Individual:** Mirroring animation on single joints/controls (eg... backs, neck, tails). This only requires inverting certain curves based on the plane you want to mirror across. For example, mirroring along the YZ plane requires inverting the X translate channel as well as the Y and Z rotate channels. In this case you could select all center joints/controls and mirror them simultaneously, or you could do them one at a time.
  By default the tool will mirror Translate, Rotate and Scale, but you can also choose to mirror only selected transforms.
- **Custom Attributes:** This allows you to also include user defined attributes such as Set Driven Key. This is on by default. If a user defined attribute exists on one side but not the other, then it will be ignored.
- **Ignore Locked Attributes:** This allows you to skip over any attributes that have been locked from the channel box. This is on by default.

Note: the local axis is used, so be sure to check orientation of joints before setting the mirror plane if you are unsure which direction to mirror.

Reverse Animation

This tool reverses all animation curves for selected objects (including hierarchies). You can reverse around specific frame, the mid-point, start or end of the animation.

For something like a mocap skeleton simply select the root transform and apply. For something like rig, you must select all node(s) containing controls.

To use: Select an animated object or node, open **Reverse Animation** window and apply with desired settings.

Time Warp Animation

This tool will create a new time warp curve for either the selected objects or a group of objects. Unlike a scene time warp, which affects everything in the scene, this can be used to only time warp certain objects, or to time warp different objects with different time warp curves. There is an additional option to attach an existing time warp curve to a new object. The tool assumes that the objects you wish to time warp already have animation curves.

If you want to effect an entire group of objects, choose the Include Hierarchy option. This option will also allow you to time warp shape nodes (e.g. a light transform plus the lighting values).

If you want to connect an existing time warp curve you must first select the objects that you want to effect, and then select the existing time warp curve (node) last before applying. Note: It may be easiest to select the existing time warp curve (node) from the HyperGraph.

Note: The resulting time warp curve will be setup similarly to a new scene time warp curve. As with a scene time warp, the initial curve will be set to linear and the start/end values will be defined by the current time range.

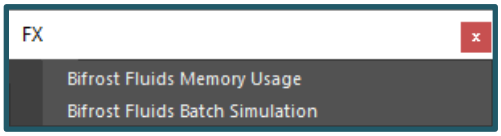
To use, select the objects that you'd like to time warp and open the **Time Warp Animation** window.

Bake Redirection Node

This will bake the effect of the legacy Redirect node on the selected object(s).

To use, select object being affected by redirection and open **Bake Redirection**

FX



Bifrost Fluids Memory Usage

This will print Bifrost memory usage information to the Script Editor output. It can be used directly while a Bifrost simulation is running. However, it can also very useful in the "Pre-Render Frame MEL" field of the Render Options. Similarly, it can also be useful for monitoring batch processing (playblasting, exporting meshes, etc). Lastly it can be handy for use in a per-frame expression.

To use, run **Bifrost Memory Usage** from the menu while a Bifrost simulation is running and check the script editor output for details. Alternate usage: simply put the command "**bifrostMemUsage**" in the Pre-Render Frame MEL field under Render Globals > Render Options.

This script contains a single global procedure named bifrostMemUsage(). When called, the current scene is examined for Bifrost containers. If one or more Bifrost containers are found, memory information will be printed to the Script Editor (or stdout, if in batch).

The printed data provides scene information such as:

- current frame
- time since last evaluation (omitted if it's the first time called)
- memory used/free

Then, for each container found, the following data is printed:

- MVS
- name of bifrostShape
- number of particles/voxels

Example output:

[bifrostMemUsage] FRAME 10

elapsed since last eval: 3.58 seconds

2019/04/08 09:44:30

Memory used/free: 0.02Gb/15.98Gb (0.13%)

bifrostLiquidContainer1

MVS: 0.5

liquidShape1

Particles: 420

Voxels: 22500

You can call this procedure in a per-frame expression to output data every frame interactively, however this may not be helpful all the time. It is recommended to call this during batch simulation or rendering by using the -preFrame flag, for example:

Render -r mr -s 1 –e 10 -preFrame bifrostMemUsage scenes/myScene.ma

The procedure bifostMemUsage() will be called at the beginning of each frame evaluation, letting you know the status of your Bifrost simulation.

Bifrost Fluids Batch Simulation

This will open a file browser to a simple set of wrapper scripts that will allow you to run a Bifrost simulation in a batch, headless mode. This enables you to cache data from the command line, without the overhead of running Maya's full UI.

bifrostBatchSim is intended to be run from the command line, either in a standard terminal for Linux/Mac or Cygwin for Windows. Cygwin, which requires installation, is a large collection of GNU and Open Source tools which provide functionality similar to a Linux distribution on Windows (see *https://www.**cygwin**.com/*)

You will need two scripts named **bifrostBatchSim** and **bifrostBatchSim.mel** (the first Python, the second MEL) work together to help you run a simulation from the command line. To use these scripts, you'll have to be able to run Python from the command line. On Linux and Mac this is built-in, but on Windows you will have to install Python, as well as a terminal in which to run the script, such as Cygwin.

Place bifrostBatchSim in your $HOME/bin folder, and bifrostBatchSim.mel in your $HOME/maya/scripts folder (or Documents/maya/scripts, if on Windows).

Once installed (and your $PATH includes your $HOME/bin folder), you can run bifrostBatchSim from anywhere. If you provide the -h/--help argument, you will get a usage message:

```
% bifrostBatchSim -h

Using MAYA_LOCATION: /Applications/autodesk/maya2019

usage:

    bifrostBatchSim [-h] --start START --end END --out_dir OUT_DIR filename

positional arguments:

    filename          Name of file to cache.

optional arguments:

    -h, --help        show this help message and exit
```

--start START, -s START

The start frame of the simulation.

--end END, -e END The end frame of the simulation.

--out_dir OUT_DIR, -o OUT_DIR

Path to output directory for caches.

There are four required arguments:

- -s/--start: the start frame (usually equal to the bifrostContainer's startFrame)
- -e/--end: the end frame of your simulation
- -o/--out_dir: the destination for the output cache files (preferably a local path with plenty of storage)
- filename: the name of the file to compute

An example command line is:

bifrostBatchSim -s 1001 -e 1150 --output_dir /usr/tmp/bifrost_sims/mySim scenes/mySim.ma

A number of things happen when you launch the script:

- the destination directory will be created if it doesn't already exist
- within this directory, a 'log' directory will be created
- the stdout of the running process will be written to a file in this log directory
- you can read this file to view the progress of your simulation

(hint: use 'tail -f <filename>' to watch the output live)

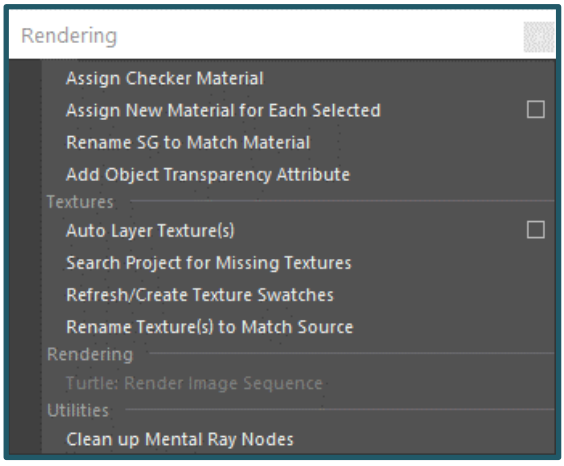
NOTES

This Python script simply performs a number of checks and runs a mayabatch command with the correct arguments, which is why you have to have bifrostBatchSim.mel in place for this to work.

This script looks for the environment variable MAYA_LOCATION, and will report an error if not found. Make sure this is set properly in your environment prior to running the script.

This script automatically adds an expression to your scene that runs an additional bonusTool script called **bifrostMemUsage()** every frame, so you don't need to add it yourself.

Rendering



Assign Checker Material

This tool assigns a new checker texture and shader to selected meshes. This tool has been modified to default to grayscale rather than green and blue.

To use, select a mesh and apply **Assign Checker Shade**r.

Assign New Material for Each Selected

This tool assigns a new lambert material to each selected object so the result will be a unique shader for each object. There is also an option to randomize the diffuse color for each material. The random range will be subtle variation of the default gray so that you can visually distinguish each object by its color.

Update: An option has been added to create Arnold Standard Surface Shaders instead of default Lambert Shaders.

Rename Shading Group to Match Material

This tool will rename all Shading Group nodes to match the names of the selected shaders. For example, if a shader is named “Concrete”, the associated ShadingGroup will be renamed ”ConcreteSG”.

Add Object Transparency Attribute

This tool creates a Transparency attribute on every selected object (meshes/surfaces) and hooks it up to its associated shader transparency. A shading switch is created so that you can have a unique and independent transparency on each object even when they share the same shader. If the shader for the object is changed, then simply rerun the tool to reattach the Transparency attribute to the new shader. The Transparency attribute can be used to gradually fade the visibility of the object either for rendering purposes or simply for display purposes in the viewport.

Update: Minor fixes for this release including renaming the custom attribute to “Transparency”

Note: This works on classic Lambert, Blinn and Phong shaders but does not work with Arnold shaders. Also, in order to view the effect of the Transparency attribute in the viewport you must be using Viewport 2.0. Shaded/textured mode needs to be enabled in the viewport (i.e. 6 hotkey)

To use, select a mesh and apply **Add Transparency Attribute**.

Auto Layer Textures

Note: This tool combines the functionality of the old auto layer tools and therefore replaces both. It will work slightly differently depending on what is selected.

If a single texture is selected it will insert a layered texture node between the selected texture and its shader if it is indeed connected to a shader.

If two or more textures are selected it will create a layered texture node and then using that node it will layer the selected textures in the order that they were selected. Changing the order of the selection will change the order that they are layered. If the first texture selected is also connected to an existing shader then the layered texture node will be inserted between the first texture and the shader.

There is now an option for including the alpha of the selected textures when layering. If this option is off, then only the diffuse will be layered. If this option is on, then the alpha of each texture will be used as a mask between the textures/layers.

Search Project for Missing Textures

This tools will search the current project for missing textures (i.e. file nodes that are pointing to an image file that doesn't exist at the given path/location.) The tool can be used in 4 different ways.

> Selected object(s) - If you select an object in the viewport or Outliner that is missing one or more textures, then apply this tool, it will attempt to find any missing textures that are associated with that object.
>
> Selected shader(s) - If you select a shader in the HyperShade or Node Editor, then apply this tool, it will check all textures that are connected as inputs to the shader. If any of those textures/images are missing it will attempt to find them in the current project.
>
> Selected textures(s) - If you select a shader in the HyperShade or Node Editor, then apply this tool… if any of those textures/images are missing it will attempt to find them in the current project.
>
> Entire scene – If you have nothing selected, then apply this tool… It will check all textures nodes in the scene. if any of those textures/images are missing it will attempt to find them in the current project.

Refresh / Create Texture Swatches

This tool will create texture swatches for file texture nodes in cases where swatches don't render by default due to the large size of the underlying image file. It will otherwise update texture swatches for any file texture node in cases where the underlying file might have changed.

Note: If one or more file texture node is selected it will update only those nodes, but if nothing is selected it will give you the option to update all file texture nodes in the entire scene.

Rename Texture to Match Source

This tool will rename one or more selected texture nodes to match the name of the actual source file. For example, a texture file node that is pointing to a file called face.tga will get renamed to "face".

Note: If one or more file texture node is selected it will rename only those nodes, but if nothing is selected it will give you the option to rename all file texture nodes in the entire scene.

Turtle: Render Image Sequence

This tool can be used to render a sequence of images with the Turtle renderer. It requires that the Turtle plugin be loaded from the Plugin Manager.

Turtle is quite fast and easy to use for high quality renders especially for effects like global illumination. However, because it is targeted primarily at rendering workflows for creating light maps and single frames, it does not allow you to batch render image sequences as you can with renderers like MentalRay. This tool will automate / batch the Turtle renderer for a sequence of frames and save them out indirectly via the Render View.

Use the standard Render Settings tab to define all of the Turtle specific quality settings, as well as any common settings like format, filename, resolution, start, end and step.

Output rendered images will be named accordingly and put in the images folder of the current project.

Clean Up Mental Ray Nodes

This tool will scan the current scene for any existing MentalRay nodes (i.e. Mayatomr) and delete them.

Misc

3DS Import Plug-in

This is a plug-in that will add a 3ds option to the regular Maya file import UI. This plug-in has not been updated in a while so it may be unpredictable.

Note: This requires the 3dsImport.mll plug-in to be loaded - see BonusTools section of the Plug-in Manager.

Cut Tiles for Image Script

This is a MEL script that will cut a specified image into smaller pieces. This script is included for command line usage. See header of script for more detailed usage info.

Here's a sample for using the command: *cutTilesForImage 3 3 64 64 file1 test1 jpg ruby;*

arguments: *int $numX, int $numY, int $resX, int $resY, string $inFile, string $outFile, string $fileFormat, string $name*

Node Created Callback Plug-in

Use the nodeCreatedCB command (created by nodeCreatedCB plug-in) to register a MEL proc to be called whenever a node is created. Registered MEL procedures should be declared to take a single string argument (the name of the added node).

This is compiled from the devkit. More information is available in the Maya technical documentation. Additionally, the cpp files in the developer kit contain useful info in the header of each file.

Warning: MEL procedures registered with this method will be called whenever a node is added to the DG. This may cause problems in certain cases. For example: each time a reference is reloaded, each of its nodes is re-added to the DG.

Flags:

-register/-r string: registers a mel callback. There is no limit to the number of callbacks that can be registered.

-unregister/-u string: unregister the specified callback.

-filter/-f string: by default, all registered callbacks will be called whenever any node is created.

Using the filter flag, you can indicate that callbacks should only be called when certain node types are created.

To determine the type of a particular node, see the 'objectType' and 'nodeType' commands.

Note: only one filter can be in affect at a time, and it will be applied to all registered callbacks.

-fullDagPath/-fdp: if this flag is specified when registering a callback, any dag node names passed into the MEL procedure will include the full dag path.

Important: The -register, -unregister, and -filter flags are mutually exclusive, only one should be used per command invocation. Do not delete the node in your callback.

Example Usage:

```
// Appends the suffix '_ply' to all created mesh nodes

global proc myCB( string $node )

{

        print("calling polyCB " + $node + "\n");

        string $type = `objectType $node`;

        if ( $type == "mesh" )

                rename $node ($node+"_ply");

}

nodeCreatedCB -register "myCB";

// End script
```

Note: This requires the nodeCreatedCB.mll plug-in to be loaded - see BonusTools section of the Plug-in Manager.

Frame Stats Plug-in

The python plugin called "pyFrameStatsCmd.py" creates a command called "frameStats" that can be used to provide frame statistics to help compare the performance of animation playback or viewport manipulation, for example in different viewports or different versions of Maya.

Usage:

```
        import maya

        maya.cmds.loadPlugin("pyFrameStatsCmd.py")

        maya.cmds.frameStats(frameCount = True)

        maya.cmds.frameStats(frameIndex = 0, frameInfo = True)

        maya.cmds.frameStats(frameIndex = 0, dirtyPropagation = True)

        maya.cmds.frameStats(frameIndex = 0, evaluation = True)
```

```
maya.cmds.frameStats(frameIndex = 0, render = True)
```

Note: This requires the pyFrameStatsCmd.py plug-in to be loaded - see BonusTools section of the Plug-in Manager. See header of python script for more detailed usage info - pyFrameStatsCmd.py is located in the plug-ins folder at the BonusTools install location.

Contributors

Through the years there have been many contributors to BonusTools. Many of these tools were developed on personal time and donated purely out of good will and a desire to make Maya ever better than it already is. A big **THANK YOU** goes out to the many people who have donated their time, energy and ideas to BonusTools over the years in one way or another. An especially big thank you to Cory Mogk who came up with the idea for BonusTools a long, long time ago. Without his foresight it would not be what it is today. There are likely more contributors and any missing names are accidental.

Mike Aquino	Tom Harper	Qi Qin
Alex Ang	Randall Hess	Jill Ramsay
Michael Ashworth	Neehar Karnik	Roland Reyer
Matt Baer	Hideo Kase	Hans Rijpkema
Michele Borghi	Tom Kluyskens	Steven Roselle
Gordon Bradley	Rick Kogucki	George Santos
Bert Van Brande	Bernard Kwok	Jason Schleifer
Duncan Brinsmead	Cheng Xi Li	Rob Skiena
Tim Brown	Lonnie Li	Jos Stam
Christine Bui	Jack Liao	Takumi Takahashi
Andrew Cammarano	Ming Mah	Ken Taki
Matt Chan	Jerome Maillot	Michael Todd
Ted Charlton	Kiernan May	Vince Tourangeau
John Creson	Jeyprakash Michaelraj	Quang Tran
Martin De Lasa	Cory Mogk	Quoc Tran
Cyrille Fauvel	Tracy Narine	Martin Watt
Tim Fowler	Rob Ormond	Changyaw Wang
Adrian Graham	Marc Ouellette	Jakob Welner
Hiroyuki Haga	James Piechota	Ian Waters
Christ Hardcastle	Kevin Pinchbeck	Joe Wu