AUTODESK **UNIVERSITY** 2015

ME11119

Creating Procedural Entourage with Max Creation Graph in 3ds Max 2016

Christopher Diggins, Principal Software Engineer, Autodesk
Martin Coven, Senior Product Designer, Autodesk

Learning Objectives

- Learn how to install and use a Max Creation Graph tool
- Learn how to open and edit a Max Creation Graph in the node editor
- Learn how to create a new procedural Max Creation Graph tool from scratch
- Learn how to use 3ds Max to create and place entourage in a scene procedurally

Description

Procedural techniques for embellishing designs with entourage can save you time by enabling you to more quickly experiment and iterate on your visualization. In this session you will learn new tools and techniques for creating and placing content procedurally in 3ds Max 2016. We will present a free set of procedural tools created using the Max Creation Graph editor in 3ds Max 2016. You will learn how we created these tools using the Max Creation Graph and how you can modify them using the Max Creation Graph node editor to address your custom visualization needs.

About your AU Experts

Christopher Diggins is a principal software developer at Autodesk in the Design Animation Group of the Media and Entertainment division. He is the architect of the Max Creation Graph visual programming language in 3ds Max 2016. He has authored numerous articles on programming languages, software engineering, and C++ programming.

Martin Coven is an award winning Visual Effects artist. In his 17 years of experience, he has worked in games, cinematics, broadcast, commercial and film. As a creature technical director for Industrial Light + Magic, Martin worked on films such as Star Wars Episode III, Pirates of the Caribbean 2, The Chronicles of Narnia, and War of the worlds. While Martin has had to wear all the production hats, he has quite a bit of focus on the animation side, including rigging, simulation work (rigid, cloth, hair and flesh) and particle effects work.

A. Introduction

Procedural Entourage

In order to make your 3D scenes more compelling, you can add "entourage" models such as plants, people, cars, street-lights, benches, and so on. These objects can greatly contribute to the immersive nature of your environments. Collectively these types of models are often referred to as entourage.

Placing entourage in a complex 3D scene can be a time consuming process for an artist. It is also very hard to iterate on in order to achieve different artistic variations, or to respond to changing customer demands.

Many tools for model placement in 3ds Max aren't procedural (for example object paint and the array tool) meaning that if anything in your scene changes, you have to manually go through and re-position the models.

In many scenarios, such as an urban environment, the placement of objects can be defined procedurally in terms of relative position to 3D objects or splines that define special interest points or roads. Doing this through constraints or controllers for each placed model,

With the introduction of the Max Creation Graph tool in 3ds Max 2016 users can now create tools that automatically place objects without having to write any code.

MCG: A Visual Programming Environment

MCG stands for Max Creation Graph. It is a visual programming environment where you can create new tools for 3ds Max by wiring nodes together in a graph. Using MCG you can make new modifiers, geometry, and controllers.

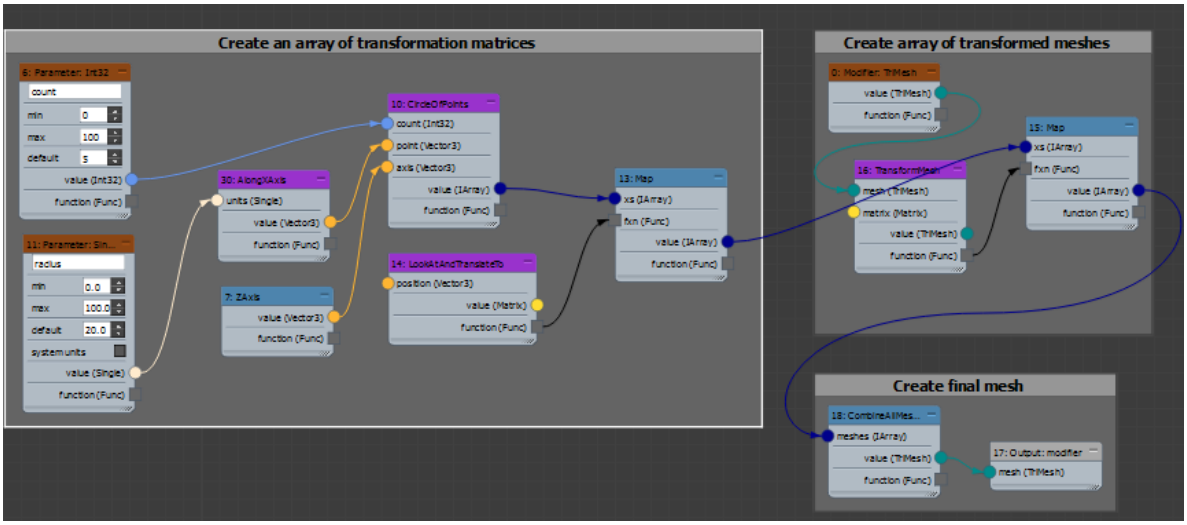


FIGURE 1: AN MCG GRAPH FOR A RING CLONE TOOL

An MCG graph is a data-flow graph that represents the set of transforms on data received from a set of inputs to the final result (e.g. a mesh for geometry and modifiers). A graph consists of nodes and connections. Nodes are instances of operators that perform transforms (summing the values in an array) or have some side-effect (e.g. printing data in the listener).

Connections represent the flow of data from the result of a computation to the input of another computation, or represent the computation itself treated as data (when using the “function” output.

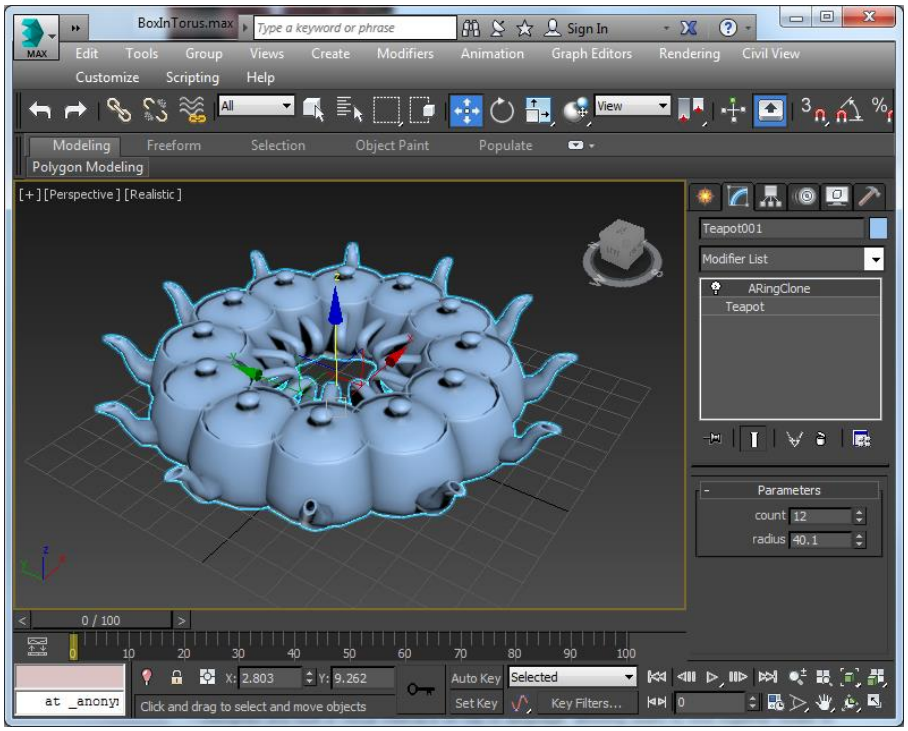


FIGURE 2: THE RING CLONE TOOL IN THE SCENE

B. Installing and Using Max Creation Graph Tools

What is an MCG Tool
An MCG (Max Creation Graph) tool is a 3ds Max plug-in (usually a geometry, modifier, controller, or MAXScript utility) built visually using the Max Creation Graph node editor. Max Creation Graphs are saved as XML files with the extension ".maxtool".

Types of MCG tools
The types of MCG tools that can be created are:

- Geometry primitives
- Modifiers
- MAXScript utilities
- MAXScript functions
- Controllers (Extension 1)

MCG Packages
MCG tools are usually distributed in package files (".mcg") which are essentially zip files. Within an MCG package is the graph XML file (".maxtool") and any compound graph XML files (".maxcompound") the tool may depend on.

Installing an MCG Package
You can install a package either from the main menu: "Scripting > Install Max Creation Graph" or from the Max Creation Graph Editor menu under "File > Install Max Creation Graph"

Once a package is installed it is accessible in different places depending on the type of tool.

1) **MCG geometry** - object creation panel; the author can specify different categories of objects, but by default they will go in the "Max Creation Graph" category.
2) **MCG modifiers** – modifier pull-down
3) **MCG functions** – accessible from the MAXScript listener or from MAXScript
4) **MCG utilities** – these are accessible with MAXScript utilities, from the Utilities panel button. Click on the "MaxScript" button, and then use the combo box to select your utiwlity and press open.
5) **MCG controller** – assign and replace controller dialogs

For a step by step walk through on installing an MCG package see the Autodesk Learning Channel on YouTube (http://autode.sk/1MTNh79).

Manually Installing MCG Tools

While the following is not a recommended workflow, understanding how to manually install an MCG tool from a package can help you understand how the system is working.

1. Create the folder "%userprofile%/Autodesk/3ds Max 2016/Max Creation Graph/Tools/Downloads/MyPackage". Where you should replace "MyPackage" with the name of your package.
   Note: "%userprofile%" is a special string recognized by Windows and will automatically expand

to your user profile in the Windows explorer (e.g. on my computer it expands to: C:\Users\digginc\)

2. Change the extension of the package from ".mcg" to ".zip"
3. Unzip the file contents to your new folder.
4. Notice you will have a ".maxtool" file in the folder and a number of ".maxcompound" files in the sub-folder.
5. From the main menu choose ("Scripting > Open Max Creation Graph Editor") to open the MCG editor
6. From the editor menu choose ("Compile > Evaluate all Tools") to re-compile all compounds and the tool files.
7. Your tool (or tools) should now be avaialbe

C. Opening and Editing MCG Tools

Opening the MCG Tool
MCG tools are stored as XML files with the ".maxtool" extension.

If you have installed an MCG package you can find the installed .maxtool file in a sub-folder of "%userprofile%/Autodesk/3ds Max 2016/Max Creation Graph/Tools/Downloads/".

If you have Extension 1 installed you can find some examples under "C:\Program Files\Autodesk\3ds Max 2016\MaxCreationGraph".

Editing an MCG Tool
The primary components of an MCG tool are the graph and the meta-information (such as description, category, etc.). The graph is shown in the main UI and has a UI that is similar to the Slate Material Editor.

Connections are made by clicking on a socket and dragging it to another node's socket. Input sockets can only be connected to output sockets and vice-versa. New nodes can be added to a graph by either:

A) clicking and dragging them from the tree view
B) double clicking them on the tree view
C) pressing X and using the search box
D) clicking a socket and dragging it into the graph without connecting it, and using the search box which appears
E) merging an existing graph

For more information see the help topic Working with the Max Creation Graph Editor: Basic Usage (http://autode.sk/1QtbEz2).

Graph Properties Dialog
The graph properties dialog lets you change meta-information about the tool such as the description, copyright, author, URL and category. For geometry tools the category is used for the Object Creation pull-down control.

Using the Custom UI tab of the Graph Properties a user can inject custom MAXScript into the auto-generated MAXScript tool. This is useful for more precise control over the UI generated for an MCG tool.

For more information see the help topic: Graph Properties Dialog (http://autode.sk/1YfAlQV).

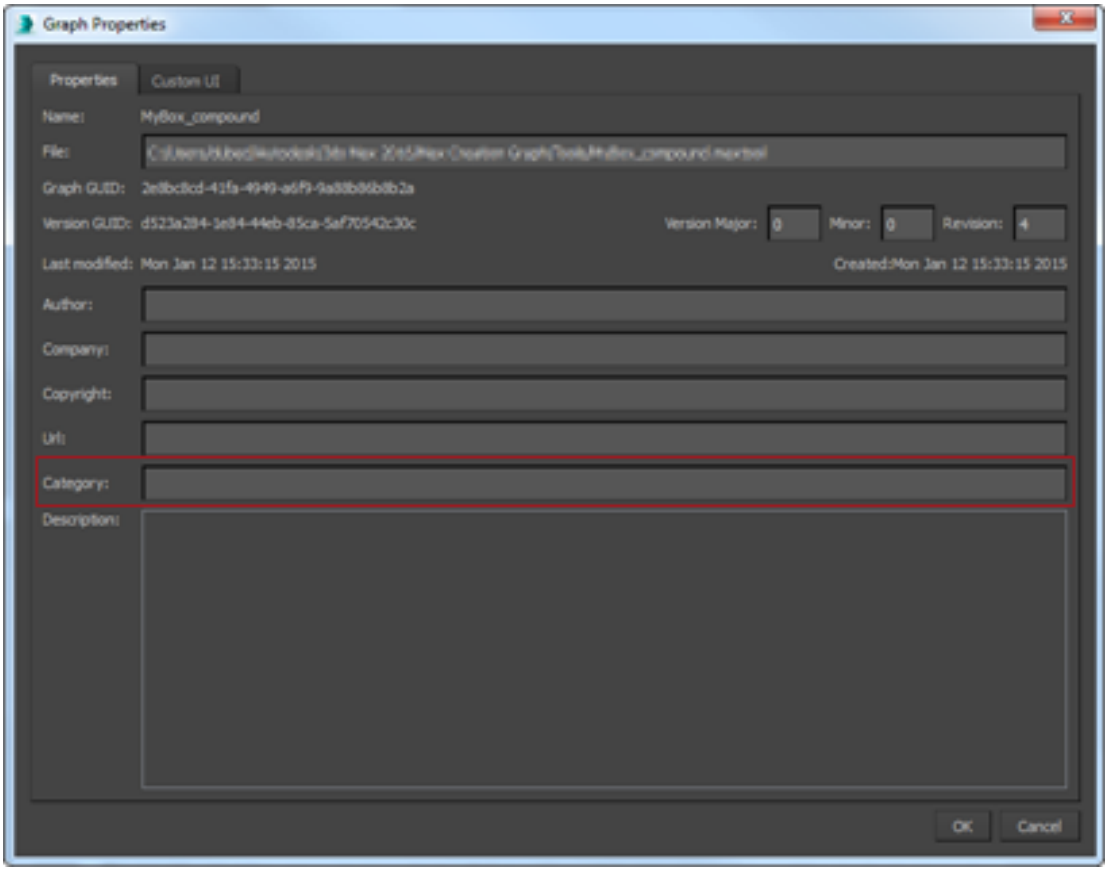


FIGURE 3: THE GRAPH PROPERTIES DIALOG

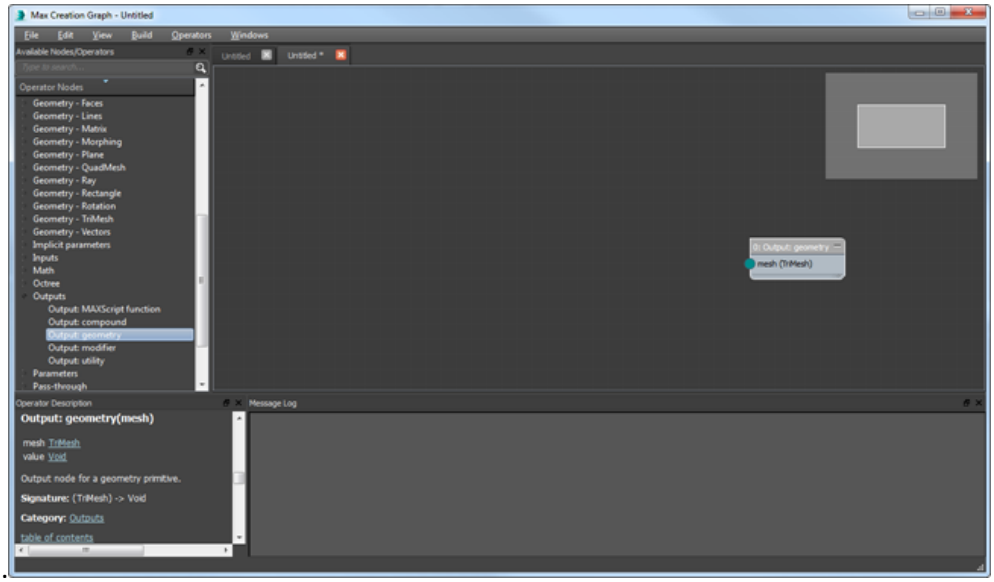
D. Creating MCG Tools

Creating your first MCG Tool

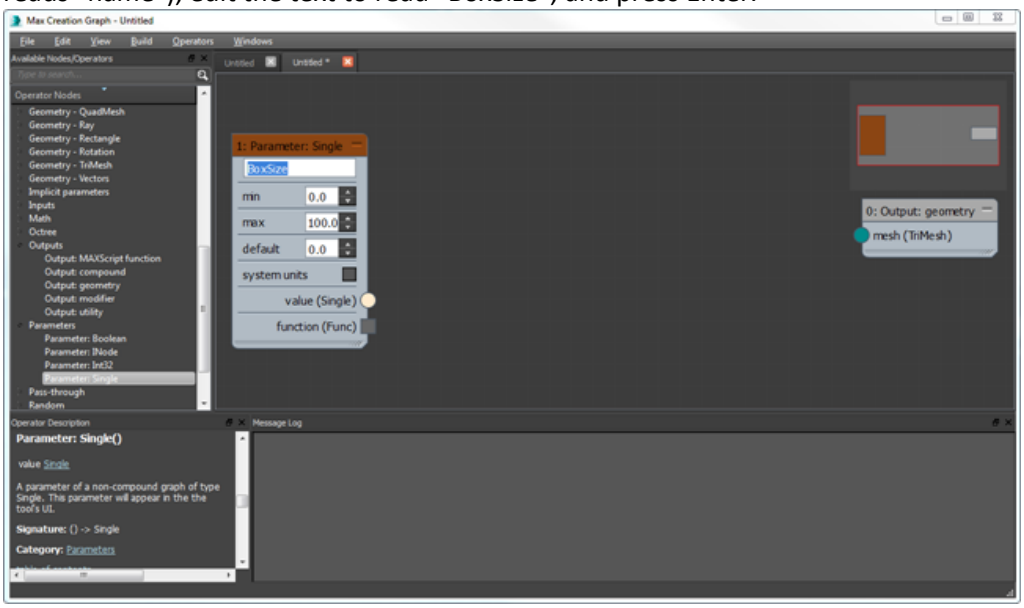
The following is a step by step walk through for creating a simple MCG geometry tool.

1. Open the graph editor: On the menu bar, click the Scripting menu to open it, then choose Open Max Creation Graph. (If using the enhanced menus, it's on the Script menu.)

2. In the Available Nodes/Operators list on the left side of the editor window, scroll to the Outputs category, expand it, and drag an Output: geometry node into the graph window, near the right side



3. Expand the Parameters category in the Available Nodes/Operators list and drag a “Parameter: Single” node into the graph window, near the left side. Double-click the text field (currently reads "name"), edit the text to read “BoxSize”, and press Enter.

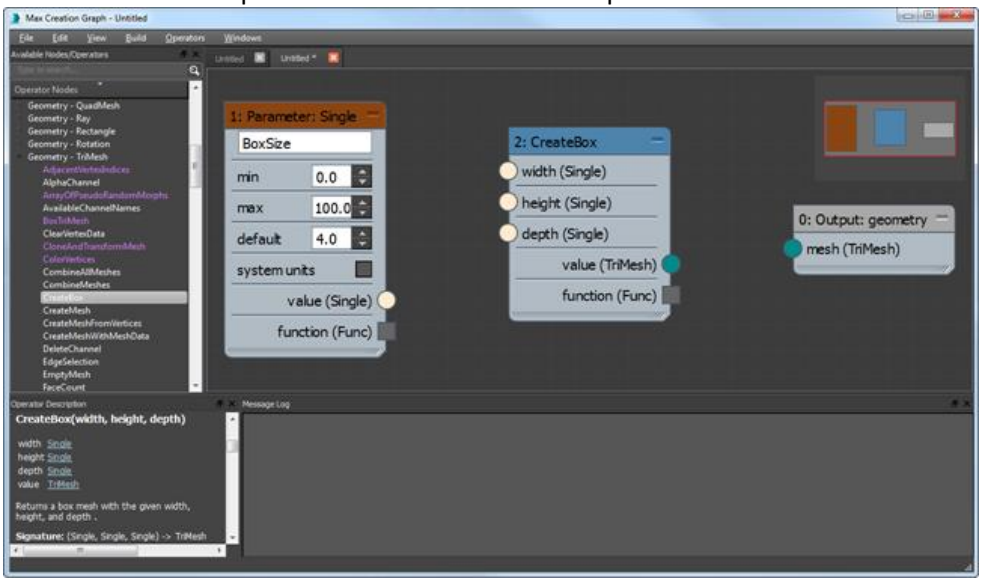


A “Parameter: Single” will create a spinner UI control in your MCG tool UI that contains a single-precision floating point value (a decimal fraction).

4. In the "default" field, enter 4.0. You can leave the default "min" and "max" values of 0.0 and 100.0.

5. Expand the Geometry - Trimesh category and drag a CreateBox node into the view. Place it between the two existing nodes.

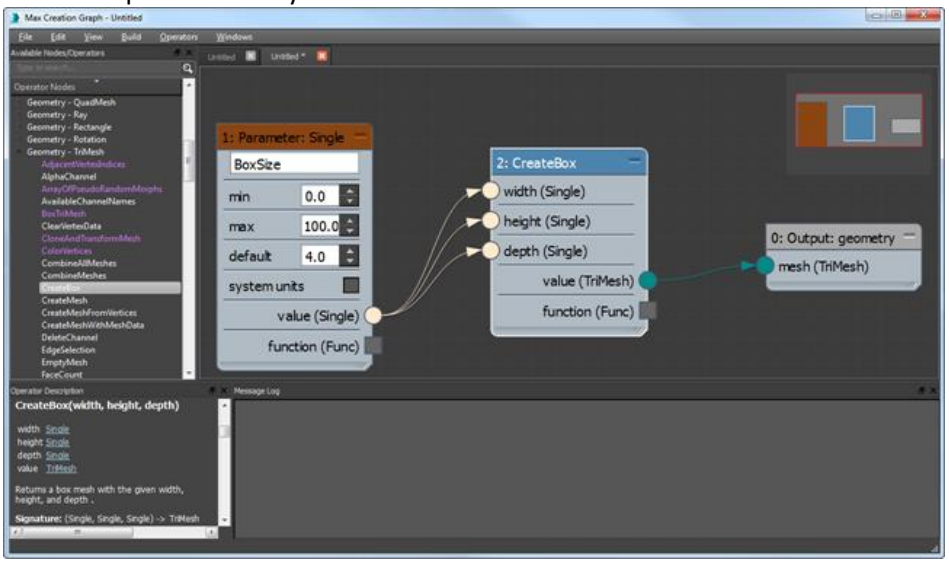
6. Connect the Parameter: Single node's output, "value (Single)", to the CreateBox node's input, "width (Single)", by clicking and dragging from one connector to the other. The direction in which you perform the connection does not affect the outcome of the graph; the data will flow from the node's output to the connected node's input.



7. Similarly, wire the Parameter: Single value (Single) connector to the height (Single) and depth (Single) input connectors of the CreateBox node.

8. The result is three wires going from the Parameter: Single node's single output to each of the CreateBox node's three inputs. This means that the one value from the Parameter: Single node will be used for all three of the box's dimensions, thus producing a cube. If you wanted to use different values, you could just add two Parameter: Single nodes and wire them separately to the CreateBox node's inputs.

9. Wire the value (TriMesh) output connector of the CreateBox node to the mesh (TriMesh) input of the Output: Geometry node.



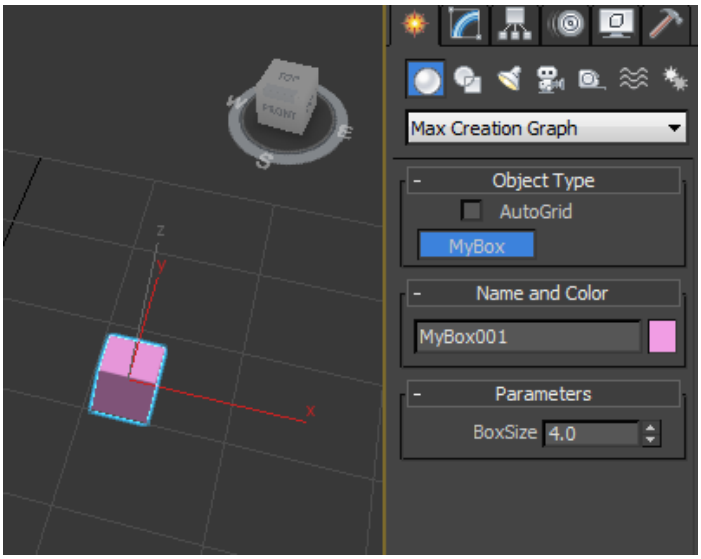
10. To use the graph output, it's necessary to first validate and then evaluate the graph. You can validate the graph manually, but saving it performs the validation automatically, which simplifies the process. Press Ctrl+S and save the file in the Tools\ folder inside the default path as MyBox.maxtool.

    The Message Log window of the graph editor shows that validation, generation of the MAXScript file, and evaluation of the MAXScript file were all successful.

11. Open the Build menu and choose Evaluate. The Evaluate function builds the graph and adds it to the 3ds Max interface.

12. Go to the Create panel, open the drop-down list and choose Max Creation Graph. Then, on the Object Type rollout, click the MyBox button.

13. Click in a viewport to add a box then adjust the BoxSize parameter value to change the box's size.



Creating a Simple Cloning Modifier

The graph in Figure 4 demonstrates a simple cloning modifier. In order to recreate it quickly you can use "X" button to search for operators. You can do this after clicking in the MCG editor view or when dragging a connection from a node. When you press "x" a list of all of the operators whose name start with the text you write will show up. You can use the wild card character "*" in your search string.

In the graph you will notice some documentation blocks called "group nodes". These are not necessary to create but it makes your graph easier to read and understand. You can create a group node by selecting the nodes you want in the group, and then selected "Create Group Node" using the right-click context menu.

> ***Tip: Auto-Generating Parameters***
> You can auto-generate parameter for graph by selecting an operator (e.g. Vector3) and selecting "Create Parameters" from the right-click context menu.

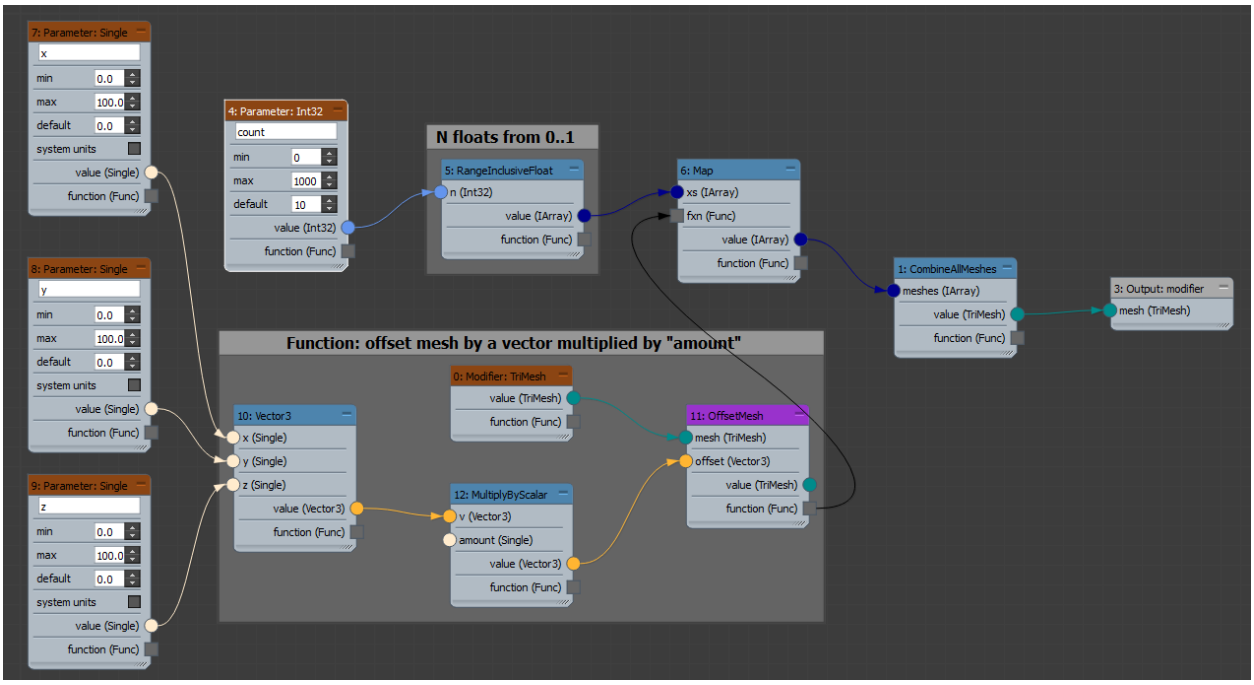


FIGURE 4: THE CLONE OFFSET MODIFIER

In the Clone Offset Modifier graph a number (specified by the parameter “count”) of decimal values are created in the range of [0..1] by the operator “RangeInclusiveFloat”. The “Map” operator creates an array of meshes by multiplying a vector by each of the values in the array and then used to create a copy of the mesh at the specified location (“OffsetMesh”).

The “CombineAllMeshes” operator combines an array of meshes into a single TriMesh object.

Once you have saved and successfully the graph (let’s call it “Clone Offset”) it is now available as a modifier tool.

Functions

Notice that in the cloner graph the “function” output socket of the “OffsetMesh” node is connected to the “Map” function input socket. This means that the left-graph starting from “OffsetMesh” (i.e. the portion of the graph leading into “OffsetMesh”) is treated as a function with the unconnected sockets acting as function arguments, in this case the “amount” pin of the multiply by scalar.

Compounds

In the clone graph you may have noticed that the “OffsetMesh” operator is colored differently from the other nodes. This is because it is defined using a graph. Operators defined in terms of a graph are called compounds. You can open a compound’s graph by right-clicking on the operator in the tree view, or on a node, and choose “Open Compound in New Tab”.

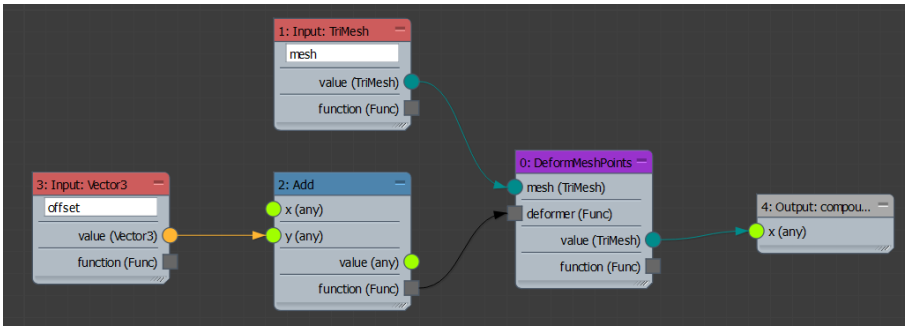


FIGURE 5: GRAPH FOR THE OFFSETMESH COMPOUND

E. Creating and Placing Objects Procedurally

Downloading MCG Sample Pack 2

NOTE: To follow along and create the graphs in this section, you will need to download and install the MCG Sample Pack 2 (http://autode.sk/1H7vXPY). Because this pack contains a large number of tools and compounds, and packages can only contain one tool at a time, it is distributed as a zip file, is needs to be unzipped manually into your Max Creation Graph folder (%userprofile%\Autodesk\3ds Max 2016\Max Creation Graph).

Placing entourage in a scene procedurally

Procedural placement means that the position, rotation, and scale of objects are determined algorithmically via a procedure that is driven by one or more data sources. The data source could be parameters in the UI, objects in the scene, or a data file (e.g. comma separated values CSV).

A few example scenarios where procedural placement tools can be useful:

1) Placing street lights along either side of a road, facing inwards
2) Placing parked cars along both sides of a road, and varying
3) Placing windows on the faces of a building
4) Creating a fence with pickets and corner posts
5) Stacking parts of a building on top of each other

When a Single Modifier isn't enough

A single object cloning or scatter modifier, no matter how sophisticated it is, is usually going to be insufficient for most real-world scenarios.

Rather than creating a plethora of specialized cloning tools using MCG, an alternative approach is to use a set of modifiers that work together by passing data through the modifier stack.

Introducing the MCG Flux Cloning Toolkit

The MCG Sample Pack 2 contains a set of tools called the "Flux toolkit". The Flux toolkit uses map channels to store object placement information in an array of matrices.

The Flux toolkit requires at a minimum two modifiers.

1) A Flux initialize modifier – Initializes a map channel with a set of initial matrices
2) A Flux evaluate modifier – Clones a mesh using each matrix in the map channel to transform it.

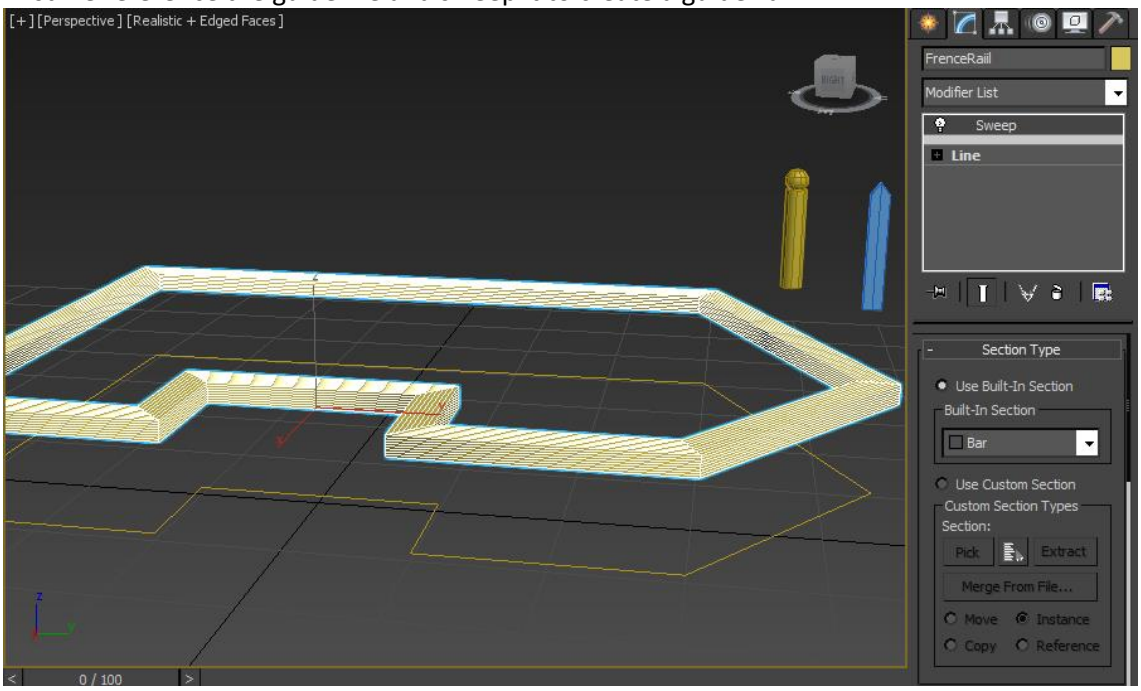
Additional modifiers in Flux fall into one of the following categories:

- Transform - perform transformations of the matrices
- Filter – removes the matrices
- Blend – modulates the effect of the top Flux effect
- Creation – creates new matrices from the current set
- Data – copies the matrix array from one map channel to another
- Meshing – creates a mesh using the matrices in a map channel

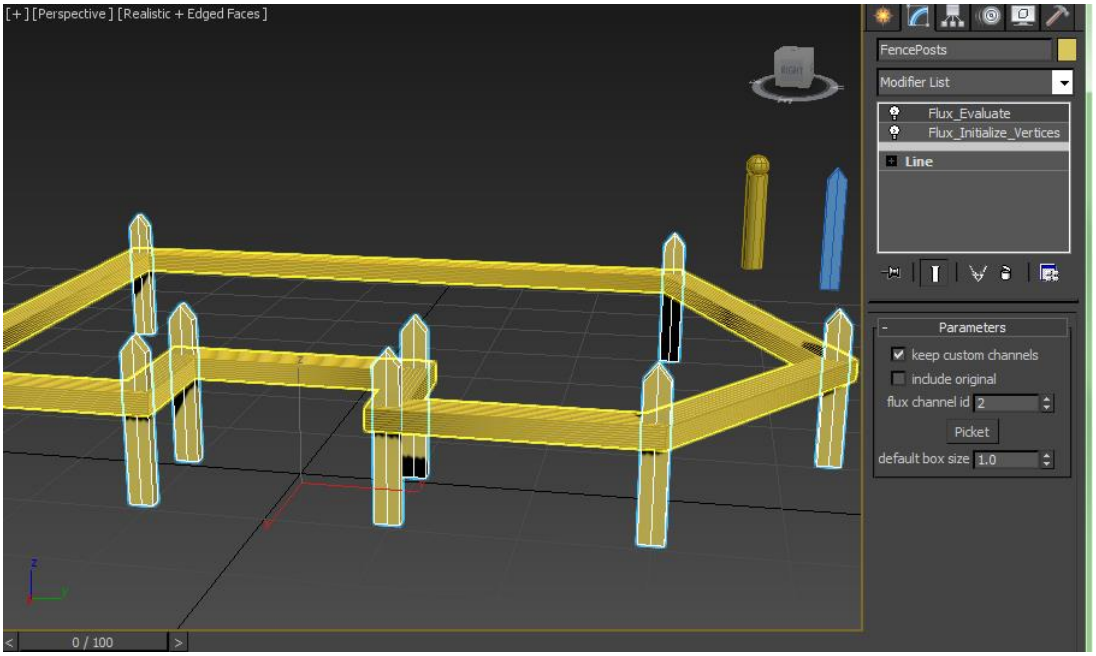
Creating a Fence Tool Using Flux

The following series of screen-shots show how a fence can be constructed from a guide spline

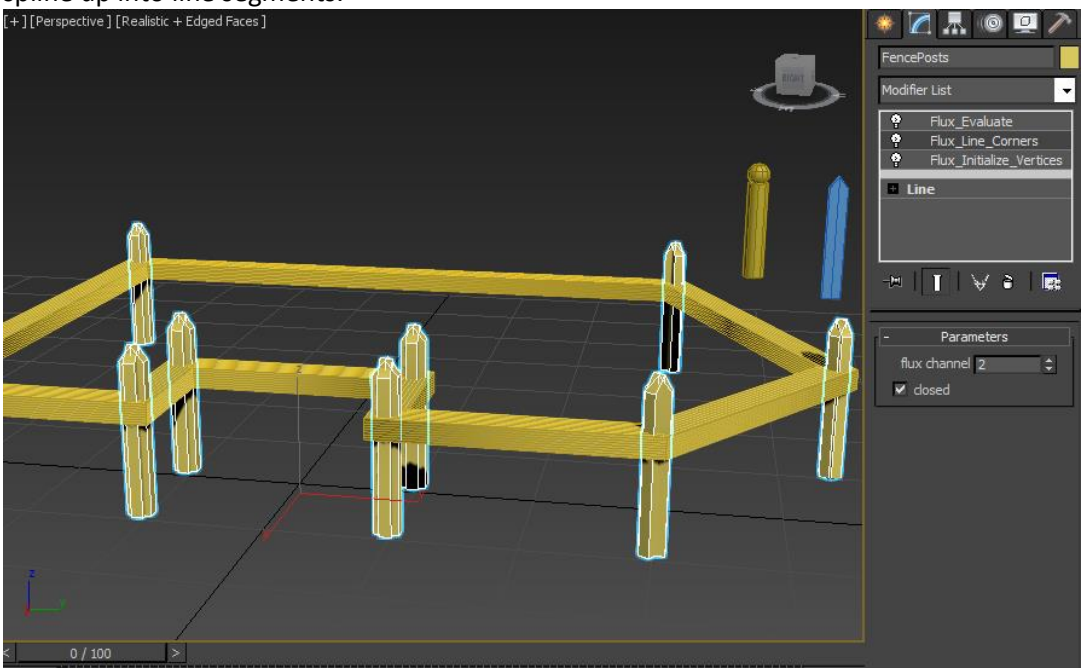
1. First we reference the guideline and sweep it to create a guide-rail.



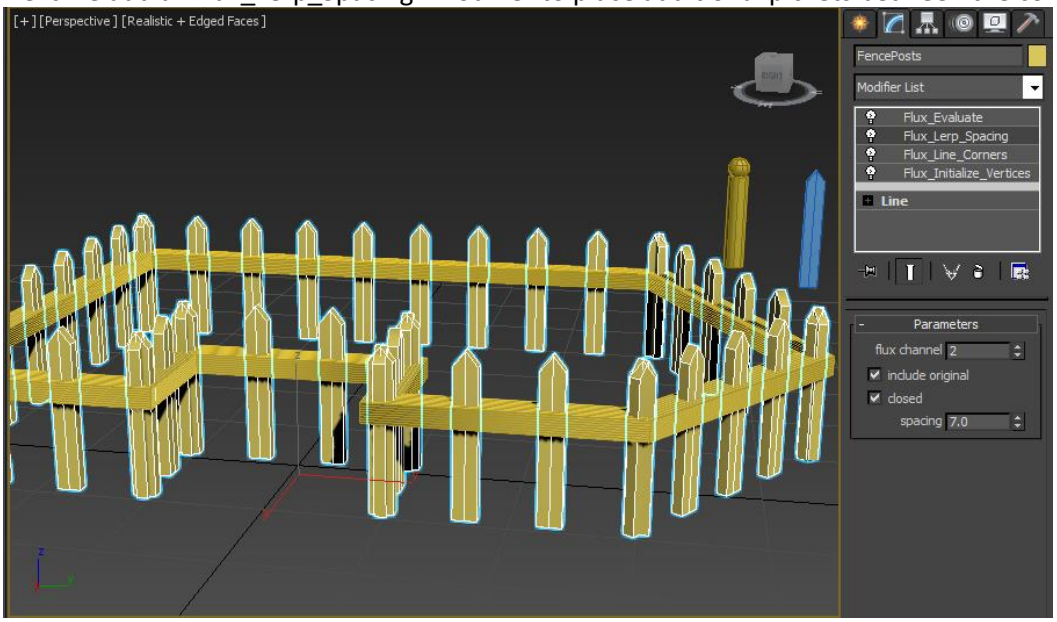
2. Next the guiding spline is referenced again. This time we put a Flux_Initialize_Vertices and a Flux_Evaluate on top which clones the picket model.



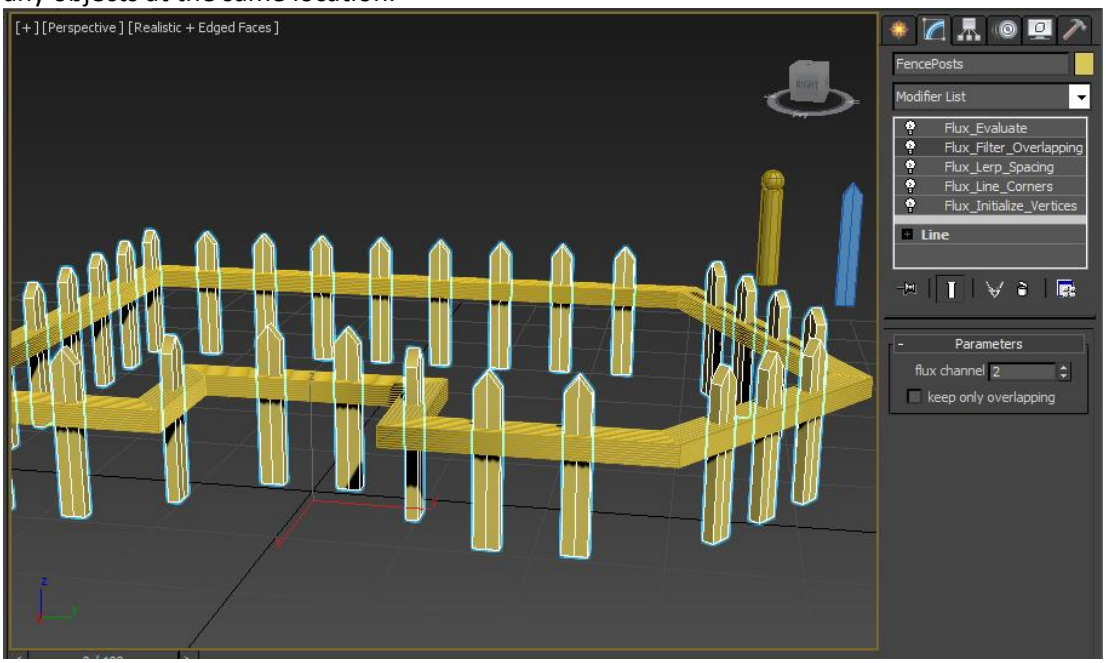
3. The next step is to add a “Flux_Line_Corners” modifier to duplicate each vertex and break the spline up into line segments.



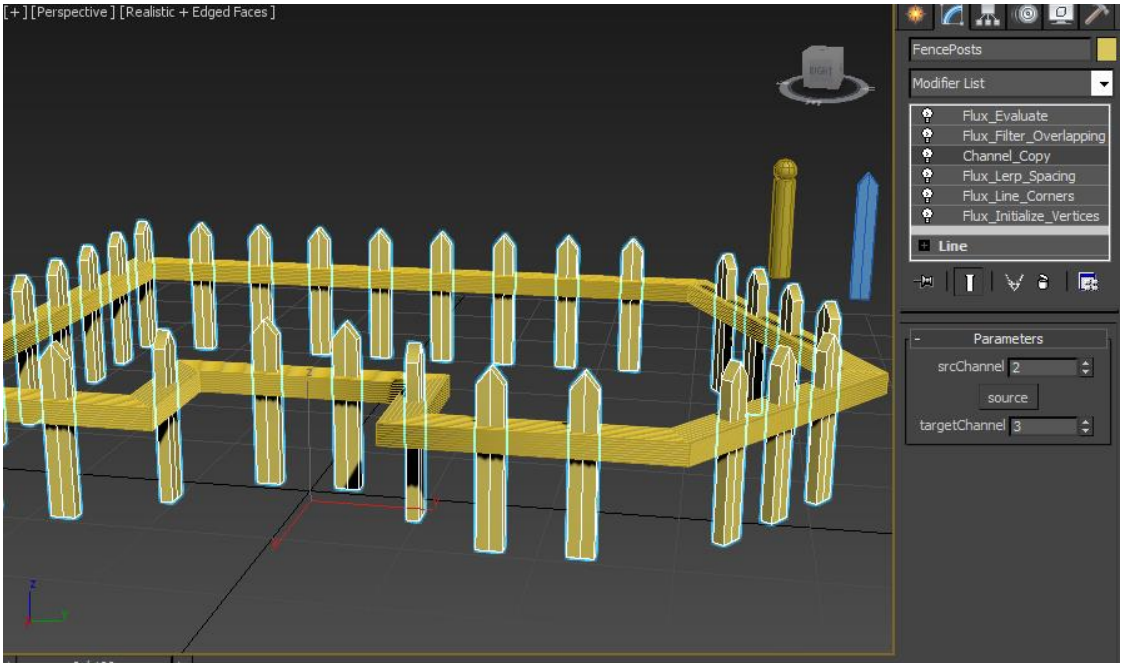
4. Next we add a “Flux_Lerp_Spacing” modifier to place additional pickets between the corners.



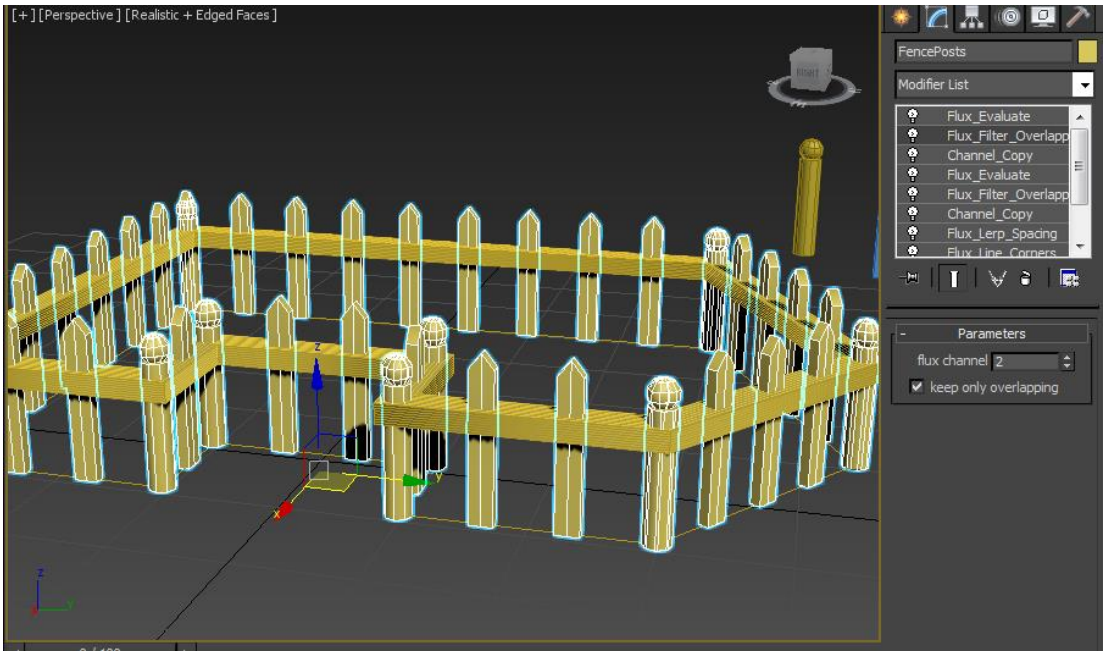
5. Now we remove the pickets at the corners using the “Flux_Filter_Overlapping” which removes any objects at the same location.



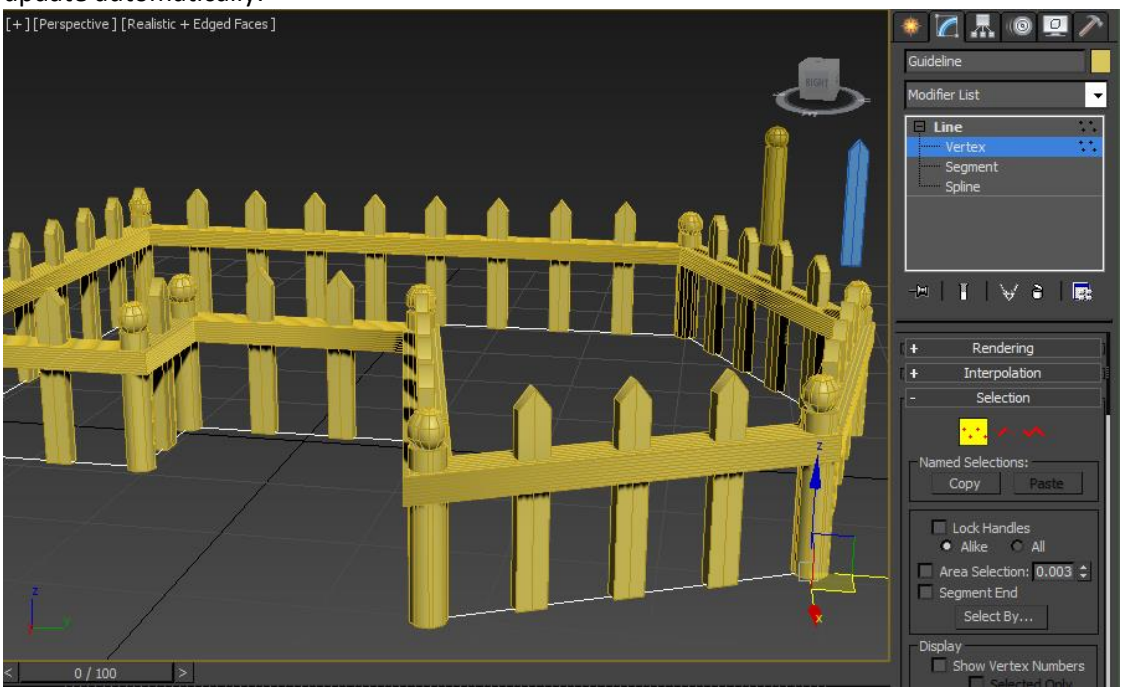
6. In order to place the corner posts, we will need to make a copy the Flux channel just before we remove the corner. We do with a “Channel_Copy” modifier, and set a temporary map channel of 3.



7. We can now add corner posts by adding three new modifiers: a “Channel_Copy” modifier to copy from the temporary map channel to the Flux map channel, a “Flux_Filter _Overlapping” with the “keep only overlapping” checkbox on, and a new “Flux_Evaluate” with the corner posts model selected.



8. Finally we have a full fence tool and can edit the guideline spline vertices and the fence will update automatically.



Anatomy of a Flux Tool

In Figure 6 the graph for the Flux_Rotate modifier is shown. This tool relies on a single compound called "FluxTransformMatrices" to do the work of:

1) Retrieving the array of matrices from the Flux map channel
2) Storing the previous effect in a back-up channel (which is the Flux channel + 1)
3) Applying the transform function to each matrix in the flux channel
4) Storing the result in the Flux map channel
5) Interpolating the current with the previous effect using the strength

This reduces the work that a Flux transform modifier has to do down to:

A) Exposing parameters in the UI
B) Defining the matrix transformation function

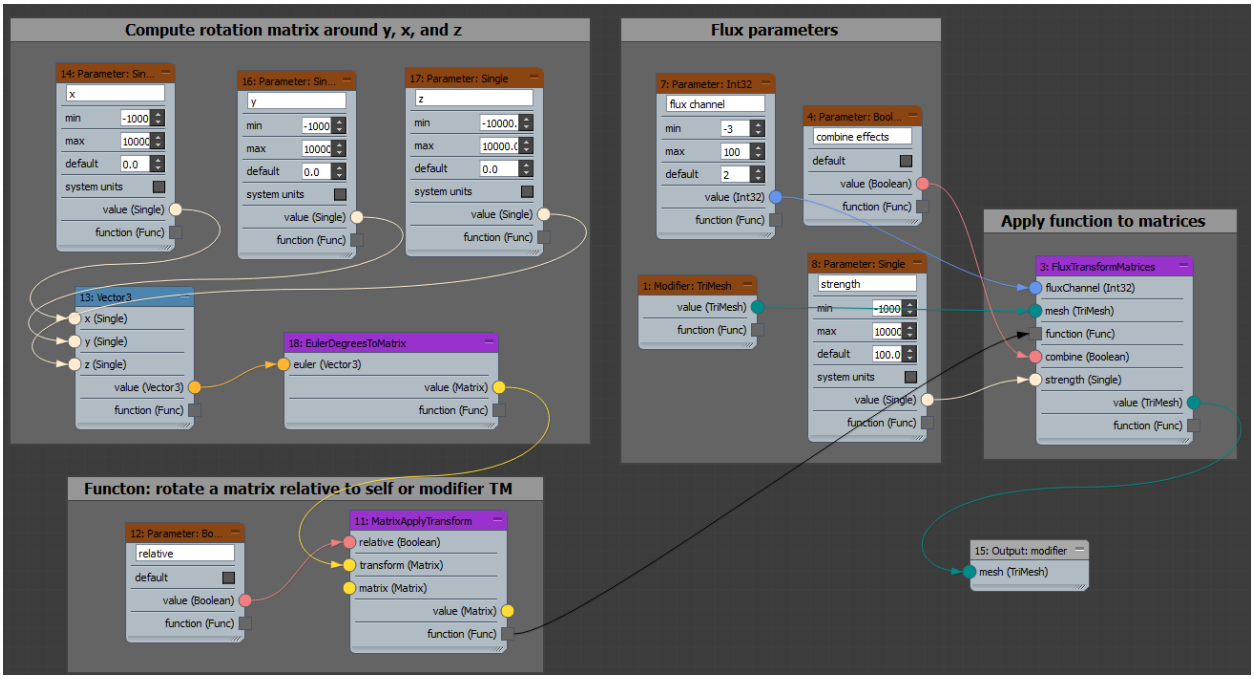


FIGURE 6: FLUX_ROTATE

Passing Data up the Modifier Stack via Map Channels

In order to for Flux tools to work together we needed a mechanism for passing an array of matrices from one modifier to another. One way to pass data along with meshes in 3ds Max is via map-channels. A map channel is a way of associating data (using in the form of points, or 3 floating point values) with particular face-vertices in your mesh. Some common examples are UV coordinates and vertex colors.

A map channel is organized as an array of data points (Vector3) and an index buffer. It is organized in the same way geometry data is stored as a vertex buffer and an index buffer. In fact in the channel info utility you can treat the geometry as a map channel (called “mesh”).

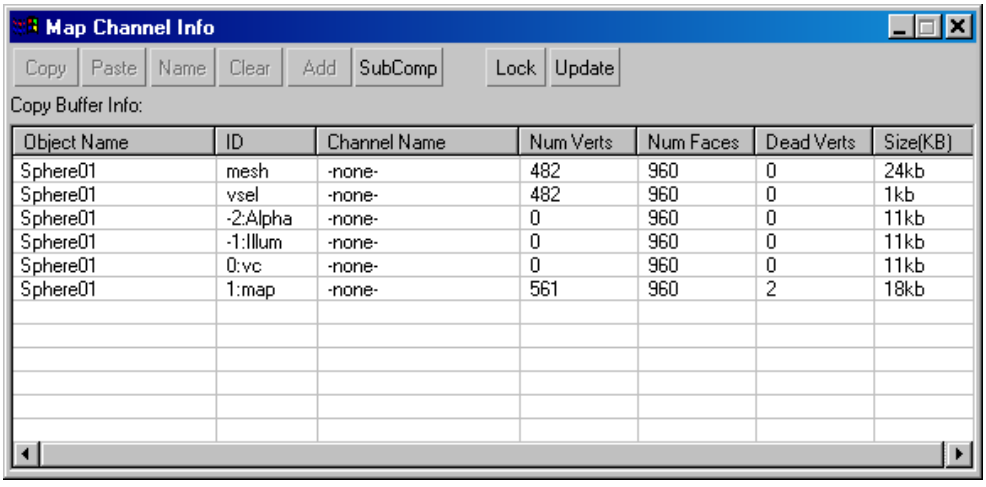


FIGURE 7: THE MAP CHANNEL INFO UTILITY

Map channels have the requirement that the number of faces (which is the size of the index buffer * 3) is the same as the number of faces in the mesh. That said, the vertex buffer can be any size. By breaking the matrix up into four Vector3 (one for each row) we can then store or retrieve an array of matrices in/from a map channel.

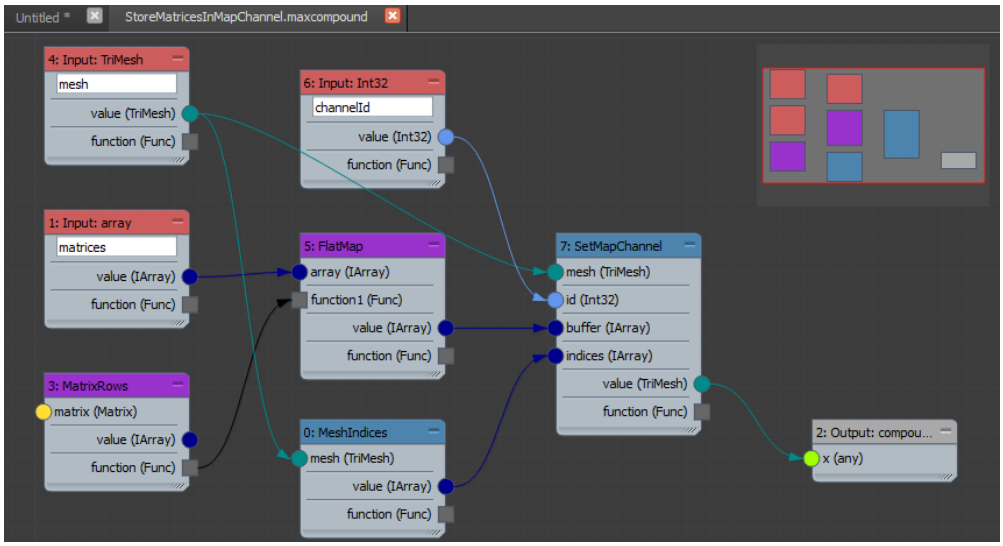


FIGURE 8: STOREMATRICESINMAPCHANNEL COMPOUND

Putting it all together

By using existing Flux tools and writing some new ones in MCG we are able to create interesting entourage placement effects for an urban environment.

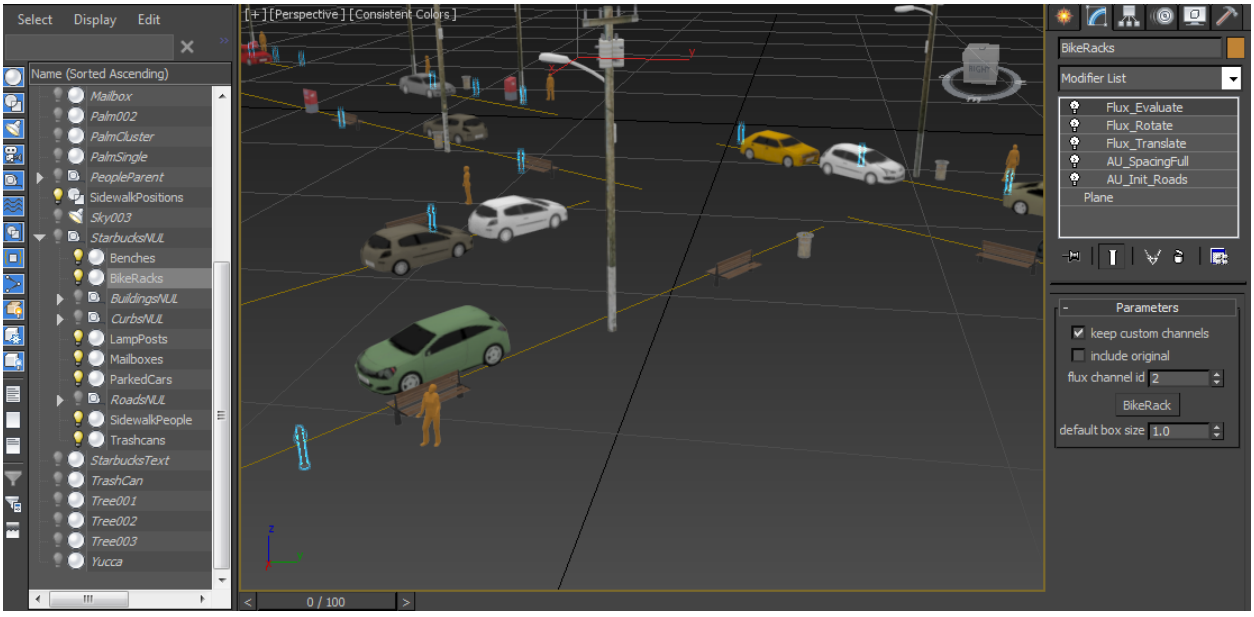


FIGURE 9: EXTENDING FLUX TO CREATE URBAN ENTOURAGE

Each of the different kinds of items in Figure 9 were created using a combination of Flux and custom tools we built for the presentation, which will be available for download from the AU site after the class and from the 3ds Max Mayhem blog (http://autode.sk/1PyQpMm).

Some of the custom MCG tools we wrote for the presentation include:

1) AU_Init_Roads – Initializes an array of matrices from a spline that define the edge of a sidewalk.
2) AU_SpacingFull – Re-distributes N objects along the spline segments
3) AU_EvaluateRandomChild – Clones a random mesh at each matrix, chosen randomly from one of the children of a source node
4) AU_RandomRotate – Randomly rotates each matrix up to a maximum amount
5) AU_RandomTranslate – Randomly translates each matrix up to a maximum amount

References

The following are links to useful resources for learning more about MCG:

- 3ds Max Mayhem Blog (http://autode.sk/1PyQpMm)
- Max Creation Graph Blog (http://autode.sk/1X5SkHg)
- MCG Facebook page (http://autode.sk/1MkY2SC)
- MCG YouTube Playlist, Autodesk 3ds Max Learning Channel (http://autode.sk/1SWMGqf)
- Max Creation Graph Help Online (http://autode.sk/1O0t195)