Using MEL to Create a More Detailed House

Starting code by Patrick Paczkowski
Additions by Yagmur Ilgen
Report by Yagmur Ilgen

Advisor: Holly Rushmeier
Introduction

Maya Embedded Language (MEL) is a scripting language used to simplify tasks in Maya, a 3D modeling and animating program. Many animators create MEL scripts to group together and speed up repetitive tasks, such as modeling various objects that might need to be created many times in a large scene. For example, a MEL script might be written to create trees in a forest. Instead of modeling each tree by itself, the animator might write a script to automatically and randomly create a tree at specific x- and z- coordinates, and could run this script until the forest is full. Of course, with a tool like MEL and scripts that fill in scenes quickly, detail and customizability are always an issue. How much of the scene can a user customize without the script getting too complicated to use? How detailed can the scene become without the script taking too much time to run? How much detail is needed to make the scene look life-like enough without having to make the script over-complicated? These questions on efficiency and detail are important aspects of modeling that animators must face with every scene they create.

This project uses a MEL script to create houses that can be customized with various details such as windows and doors. All of this can be done in Maya itself, but many users find the interface difficult to use. For this reason, I have tried to make this script much easier to use, so that a novice could come in and easily create a complex and detailed house.

Research

Before I began writing the code to create this script, I had to learn how to use MEL. I read *The MEL Companion: Maya Scripting for 3D Artists* by David Stripinis which taught me how to use
MEL to create scripts and write code to create polygons and NURBS. Once I had understood how MEL worked to create objects in Maya, I started playing around in Maya. A very useful tool was going into Maya, creating an object through the normal GUI, and reading the corresponding MEL script. After playing around sufficiently enough to have a solid handle on creating objects, I began to read through the existing code.

I carefully read through all of Patrick’s code before beginning to code my own additions to his script so I would know exactly where each function was called and how the current buildings were created. I encountered a division by zero error when I ran his script, but it was easily fixed once I read through and found out where the error was.

In addition to reading through the code, I also ran the script numerous times in variable ways so that I could get a handle on how the windows worked, and so I could compare code to the actual output of the script. The original script created a simple cityscape with residential, commercial, and skyscraper building types. Once I was familiar with the script and the code, I began to work on my additions to the project.

Additions

Instead of creating a large and less detailed cityscape, I decided to focus on just the residential building aspect of the script. Professor Rushmeier and I discussed the difficulty artists face in not being able to customize 3D models very easily because of the complicated menu system Maya utilizes. Thus, it was decided that I would create a simpler version of the Maya menu screen that would create a customizable house.
I began by extracting out all the code about the commercial buildings and skyscrapers. I decided this code was not relevant since I wanted to focus only on the residential buildings. Then, I proceeded to create sketches of windows and doors I would like to incorporate in my project. I also began to work out how I would create the windows the user would encounter when running the program, and how I should arrange the input variables needed from the user.

The following images show the various windows and doors I created during this process:

A rectangular window

A half circle window

A square window

Normal circular window
The windows and doors all have detailed protrusions which make them more realistic and more desirable to artists looking to create a detailed 3D model. The circular windows, for example, change in the scale of the outer rim. This offers the window a different appearance based on the users desires. It would be very useful to come up with a large database of windows, doors, and other details for the user to choose from, but based on time constraints and the complexity it would create for the script, I decided to simplify my database to these examples.

In the end, when the program is run, the user is capable of creating a large number of unique and detailed houses, from circular to triangular. I have created some examples, from simple versions depicting each of the windows and doors, to a much more detailed house that combines all the elements in the script.
This image shows the Maya window I created to let the user choose which size of circular window he would want to have.
In this image I have created one size of the circular window, and am now creating another size.
This is the final product displaying the arched door, circular bay window, and two different sized circular windows.

This image shows the beginning of the script, where the user is asked to choose the size of the plot of land where the house will be built.
A new type of house, this circular building displays the rectangular door and multiple half-circle windows. There is also a chimney on top, which is another detail I added to the project.
This simple construction showcases the rectangular bay window, square window, and rectangular window.

These next few images display a more complex house. This house was actually created by placing an L-shaped house onto a rectangular shaped one, and then all the details were added. It showcases every type of window and door in the program. The images offer various viewpoints of the house, so the reader can see how all the elements work together.
Along with the actual windows, doors, and chimney I added for the user to detail the house, I also added GUI windows to make the users experience a little easier than if he had used the Maya menus. Instead of having the user draw each polygon cube for each window, he is able to create a window with the click of a button.

Considerations

When I was designing this project, I ran into some issues that I had to take into consideration. The main issue I came across was the question of how complicated I wanted to make my GUI windows. There is a tradeoff between the complexity of the windows and how customizable the script becomes; for every customizable feature added to the project, I had to complicate the windows. For example, to allow the user to pick various different sizes for his circular windows, I had to create a whole new GUI window to take care of the input from the user. After realizing this, I decided to keep it simple, but allow the user some freedom over what the windows and doors look like. However, if I had more time, I would try to resolve the tradeoff and create easy to use menus that allow a user to manipulate and create highly customized details.

I also had to limit the number of windows and doors that could be created, for the reasons mentioned above. The more choices the user has, the more complicated the GUI becomes. Again, if I had more time, I would look into resolving this issue so that the user can choose from a larger database of details.

Using the Script
To use the script, download the .mel files and save them to the maya/2008/scripts folder (the year might be different based on the version of Maya). After opening Maya, open the script editor and load and execute each script file one by one, saving launchBuilder for last. When the program runs, you will be able to create a detailed house!