Wedgie

Date: December 6, 2016

By: Suzanne Berger, zanefx7@gmail.com

Introduction

This describes a Maya tool called a *wedgie* that systematically varies selected node attributes so that iterative evaluation can be done efficiently. The software is currently a work in progress. It can be considered a functional prototype but is not ready for distribution. This tool is based on concept of a color gamma wedge, a process done in the early days of film when multiple plates needed to be matched using optical printers.

Top Level Software Architecture

1. Calculation of wedge data.
2. Graphical User Interface (GUI).
3. Display of wedge data.
4. Render of wedged scenes.

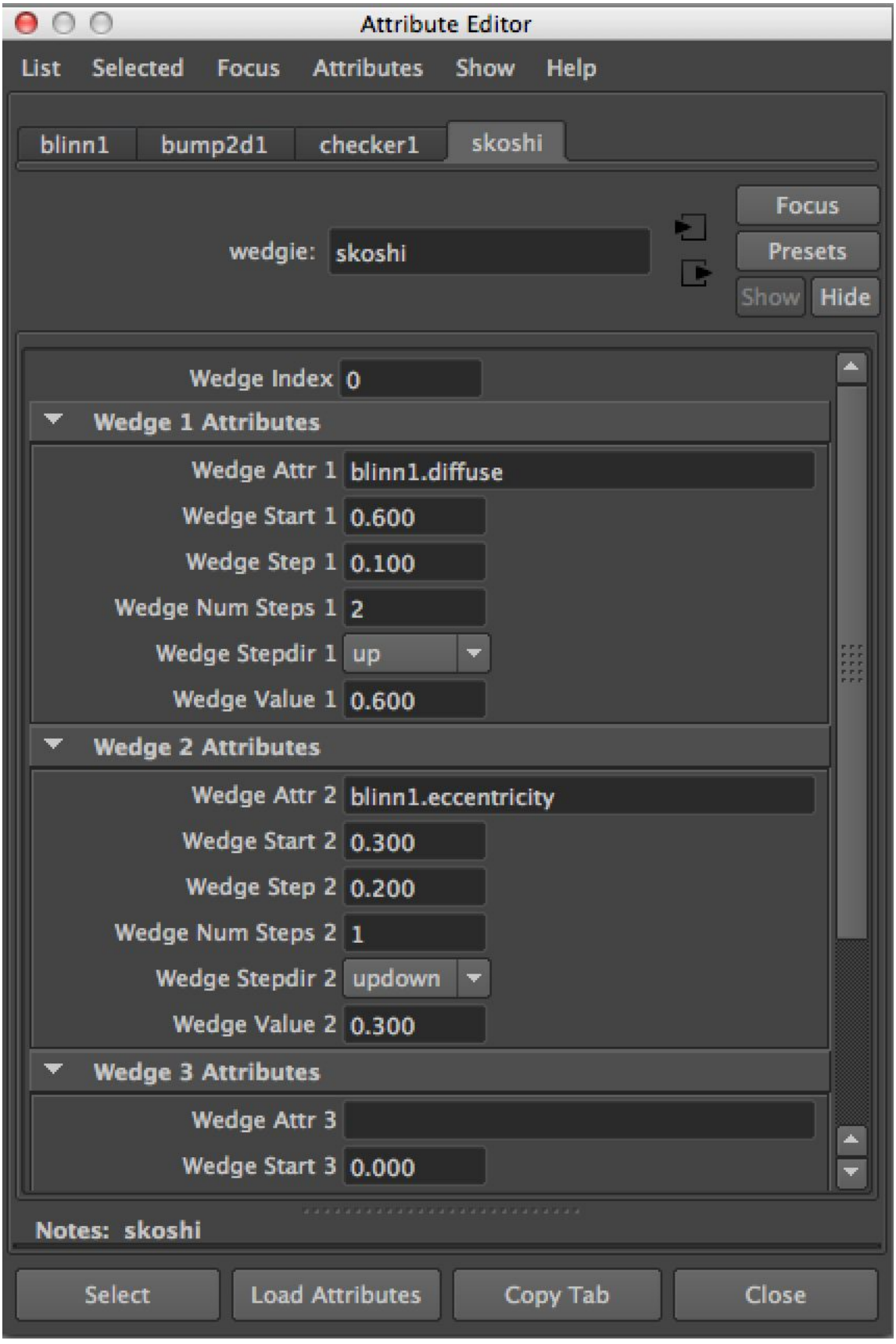
These components should segregate logic as much as possible and be contained in separate modules. For purposes of speed optimization and product distribution the first two items, should be implemented in C++. Items 3 and 4 could be implemented in Python and it should be easy for customers to integrate their own versions into the *wedgie* system.

Calculation of wedge data

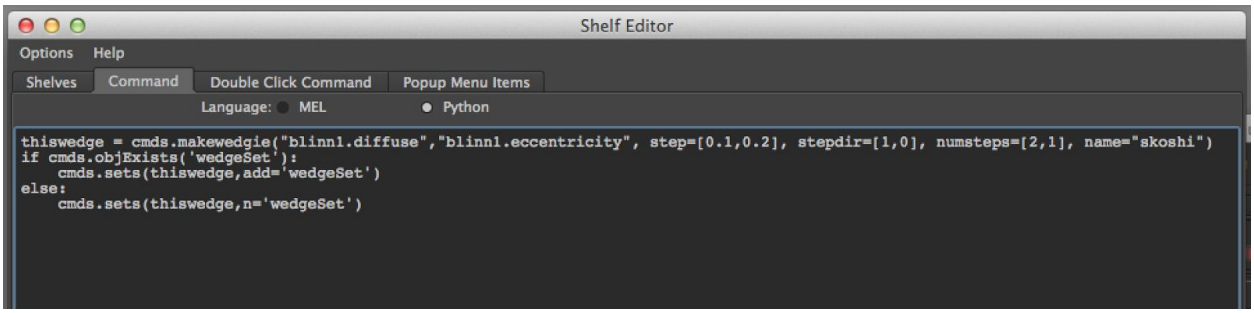
A Maya plugin is being developed using Python and OpenMaya, the Python wrapper around the Maya C++ SDK. Ultimately this will be translated to C++ for speed and distribution advantages. It consists of both a custom registered *wedgie* node and registered command, *makewedgie*, that creates the node and builds the wedge data. A technical animator from Disney recommended that a standard Maya expression node be used instead of registered custom node. This is being considered. However, implementation of custom command that calculates wedge data should remain. If expression node approach is used, then a callback system would be needed.

Note: Need to document clearly how wedge data calculation works. Ranges for each attribute is based on start value, step size, number of steps and step direction. The ultimate array of values over all wedged attributes provides every combination of each value for each attribute being wedged. It's possible that user might only want the ranges of values to be stacked. One way to do this would be to have multiple wedge nodes each with only one wedged attribute. However we might consider stacking values as an option for wedgie calculation in the node itself. ***To Be Discussed.***

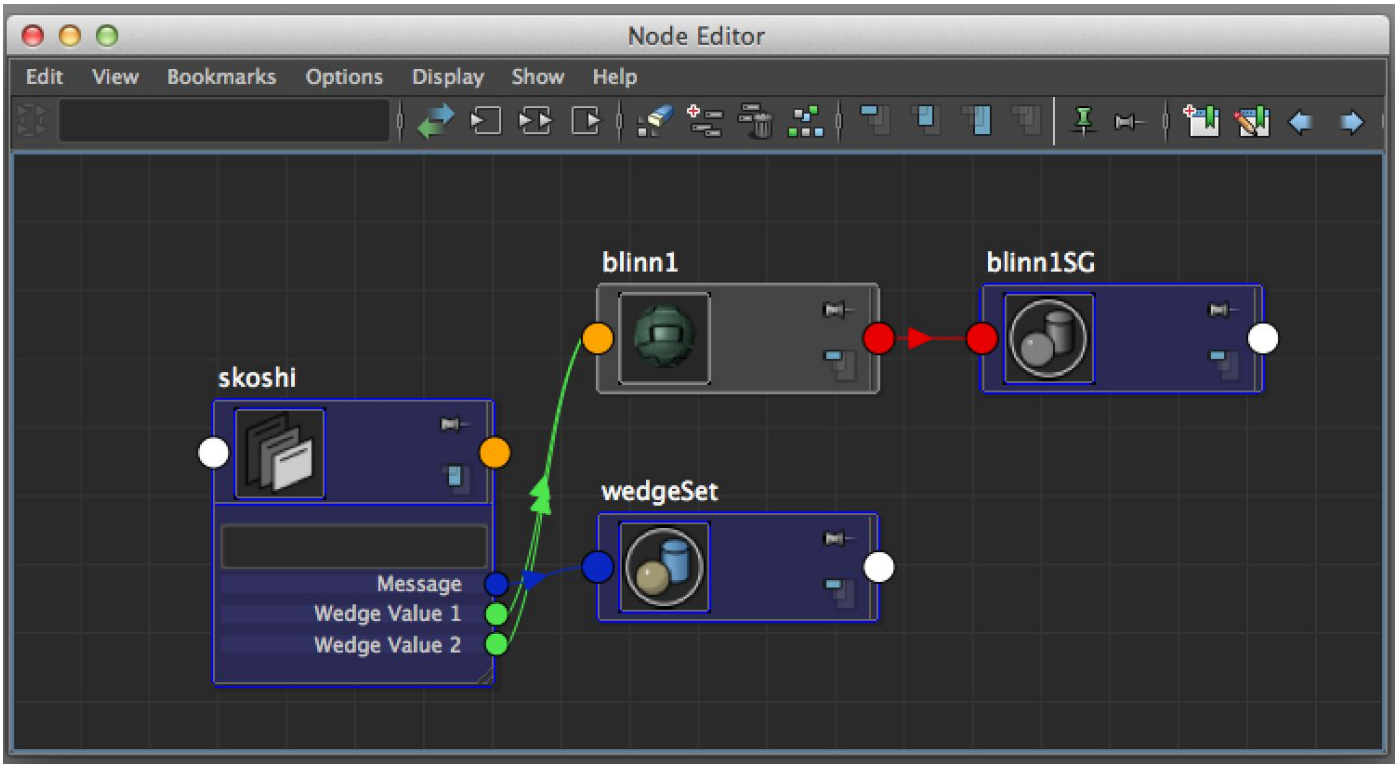
Here is example wedgie node displayed in attribute editor:



This is the command that created above wedgie:



Here is example wedgie showing its connections in Maya's Node Editor:



Note that the separate wedged attributes do not need to be part of the same node. In another example a wedge was created on particle animation using the particleShape's inherit factor attribute and a turbulence force magnitude attribute.