

# Blender as an Educational Tool

Jason van Gumster

Virginia Commonwealth University and Your Mom Design & Development

## Abstract

With the ubiquity of technology increasing, teaching artists has become more multifaceted. Art students are typically taught to think very openly and conceptually. While extremely beneficial, a problem arises when graduating artists have no developed means of analytical *problemsolving*, putting them at a disadvantage when dealing with the technology they use to create their art. It is critical that computing courses that include programming be introduced early in an art student's career, exposing them to the knowledge of how their tools work. A solution to this is to use the open source 3D modeling and animation suite, Blender, and its integration with the Python scripting language as a means of teaching this valuable way of thinking to artists.

## 1 Introduction

Over the years, as the world has become increasingly technologically-inclined, the problem of teaching artists has become respectively more complex. As has always been the case, art students are taught from the beginning to think very openly and conceptually. Heavy emphasis is placed on abstract conceptualization and any analytical thought is weighted on problem discovery and discussion rather than problem resolution.

The importance of this aspect of thinking cannot be downplayed, especially in an art field. Many an engineering and computer science curriculum would benefit from even a modest introduction to this way of thinking. However, the problem is that many of the artists presently coming out of school have little to no means of thinking analytically about problem solving. Many tend to be poor troubleshooters and tend not to approach a problem piecewise or algorithmically.

This is not an overly serious problem, but it puts artists at an incredible disadvantage when it comes to dealing with technology, even more so for computer graphics artists and animators. Their workflow is impeded because they have to spend an excessive amount of time figuring out how certain software works or trying to recover from an innocent, but avoidable mistake. Students that do not have this problem typically have worked with computers before or have taken some initiative to learn basic principles outside of the regular coursework.

Based on this, it is critical that some kind of technical computing courses be introduced early in an art student's educational career, particularly in a university setting. These courses must be something more than learning how to use industry accepted graphics applications. They should include some basic programming or scripting, exposing students to some of the inner workings of the applications they use. The more they can "think like a computer," the more these students can wield these technological tools as experts and true artists.

## 2 Purpose and Premise

The proposal here is to use the open source 3D suite, Blender, as a platform for such a series of courses. Blender is a professional-

level 3D modeling, animation, rendering and post-production solution. It features a built-in scripting and game creation engine based on the Python scripting language. Furthermore, Blender was designed to perform on a wide range of software and hardware platforms including Windows, Linux, Irix, Solaris, FreeBSD, and OSX.

Blender is an attractive solution for two major reasons: the integration of the Python scripting language and its cost. Python is an open source interpreted programming language that is quite possibly one of the easiest languages to pick up by novice programmers. Its clear, object-oriented syntax provides a powerful interface for controlling objects in Blender as a means of everything from building models to scripting animations to building fully functional video games. This quickly allows students to familiarize themselves with the concept of "thinking like a computer" while instantly putting what they learn to practice and seeing the results instantly.

The second most attractive aspect of using Blender is its cost. There is none. Being open source and freely downloadable from the internet, students have greater access to the application and more time to learn. There is no time wasted waiting for lab time at schools with limited resources for purchasing computers or the licenses to run commercial 3D software on them. Students can work and learn at home on practically any recent computer they own for the cost of a burnt CD or the bandwidth to download the application off of the web.

The advent of new technology has brought incredible potential for creative expression in digital media. With intelligent choices in curricula and the means to teach said curricula, this potential can be fully realized.