



GIS with Unity and Unreal Engine

Comparing game engines for visualizations and simulations of GIS data

Introduction

With 3D GIS data becoming more common in industry, the importance of game engines is increasing for creating high quality geographic visualizations and interactive simulations. Unity and Unreal Engine are two game engines well positioned for these needs. This document compares the two engines to help evaluate their suitability for GIS visualization and simulation tasks. The workflow for creating animations and interactive simulations of models generated by CityEngine, Esri's procedural modelling software, will be explored.

Unity and Unreal Engine

Unity and Unreal Engine are game engines that can be used to create 3D applications and animated videos. Both engines support output to multiple platforms, including mobile operating systems such as iOS and Android. They both have similar user interfaces, as can be seen on the title page image with Unity on the left and Unreal Engine on the right. The workflow for importing content into the engines is comparable, involving the import of assets into a project's library and positioning these 3D objects along with cameras, lights, and character controllers in the scene.

The game engines provide tools for animating objects in the scene along with a physics engine, which can be used to simulate behaviour such as object collision, gravity, and character movement.

The similarities diverge when looking at customizing the engine to create new gameplay elements. In Unity, C# and JavaScript are the supported programming languages, while Unreal requires either using a node based visual editor called Blueprints to link engine functionality or programming in C++.

Although Blueprints provide a way to create custom behaviour without programming, the overall learning curve of Unreal Engine is steeper than Unity due to the user interface being more complex.

Data Workflow

Unity

CityEngine can import FBX, DAE, 3DS, DXF, OBJ, and SKP files into the engine.ⁱ When exporting from CityEngine to Unity, using a binary FBX export is the recommended format as it reduces file size and loading time.

Unreal

The model import type that Unreal Engine supports is currently only FBX.ⁱⁱ However, a new version of Unreal Engine called Unreal Studio, which provides the ability to import data directly from supported design software, is currently in beta release. Recent releases of CityEngine support exporting data into Unreal Studio's Datasmith file format. A CityEngine Unreal Studio template is also available that contains materials that translate the CityEngine material attributes to Unreal materials.

Comparison

The 3D model export from CityEngine to the game engines can be time intensive, especially if the process is executed frequently due to model updates. The table below summarizes a sample export of over ten thousand tree models generated in CityEngine to the game engines.

| | Unity | Unreal Engine | Unreal Studio |
|--|------------------------|------------------------|------------------------------|
| Export of tree models from CityEngine | 30 seconds (FBX model) | 30 seconds (FBX model) | 10 seconds (Datasmith model) |
| Import of tree models into game engine | 8 minutes | 8 minutes, 30 seconds | 10 seconds |
| Model File Size | 34.2 MB | 34.2 MB | 46 MB |

The import process into Unreal Studio was much quicker because instanced meshes are supported with the Datasmith export. This instancing works by referencing duplicate geometry instead of exporting every instance individually.

The export of 3D buildings was also timed to evaluate the performance using models that aren't suitable for instancing due to non-repeating textures and geometry. As expected, there was no speed improvement for Unreal Studio over Unity. However, it was surprising how much quicker the FBX export executed compared to the Datasmith export.

| | Unity | Unreal Engine | Unreal Studio |
|--|------------------------|------------------------|---------------------------------------|
| Export of building models from CityEngine | 14 seconds (FBX model) | 14 seconds (FBX model) | 1 minute 15 seconds (Datasmith model) |
| Import of building models into game engine | 2 minutes 55 seconds | 9 minutes 48 seconds | 3 minute 38 seconds |
| Model File Size | 18 MB | 18 MB | 33.4 MB |

Additional Assets

Unity Asset Store

Unity provides an online store where assets can be purchased or downloaded for free. These assets include 3D animations, models, shaders, and scripts.

Unreal Marketplace

The Unreal Marketplace can be used to purchase assets such as animations, characters, music, textures, materials, and plugins. Some of the assets are also available for free download.

Comparison

The ability to easily find and download assets can be incredibly useful to save time when working on a 3D project. Unity seems to have more assets available for download. For example, the Unreal

Marketplace returns 11 results when searching with the keyword “traffic”, while Unity returns 392 results. The stores can be searched online at the following addresses:

Unity Asset Store- <https://assetstore.unity.com/>

Unreal Marketplace - <https://www.unrealengine.com/marketplace/assets>

Animations and Simulations

Unity

Imported models can be explored interactively by adding a first or third person game object to the scene. Alternatively, the built-in animation system can be used to move the camera through the scene for a fly through.

In Unity, there is no built-in mechanism for saving the output of the game view to a video file. However, free packages are available to download from the Unity Asset Store to accomplish this task.

For simulating character movement on a surface, a Navigation Mesh or NavMesh can be built based on imported geometry such as a model of sidewalks or paths. Once characters are configured as NavMesh Agents and assigned a script that determines the destination goal, they will be able to find the shortest path on the NavMesh while avoiding obstacles.

Unreal

Unreal Engine offers the ability to render movies created in the sequencer, a multi-track editor for manipulating the transforms of game objects as well as transitioning between camera views.

To create an interactive application where the camera can be moved through the scene, Blueprints are needed to map the control keys to the camera movement. Unreal Engine provides template projects which include useful game objects such as first and third person character controllers to speed up the development process.

Unreal supports building NavMeshes for character movement. Once a character is assigned an AI Controller created with either Blueprints or C++, it will be able to navigate to destinations on the NavMesh.

Comparison

Creating a simple character controller can be accomplished in both Unity and Unreal, though more steps are involved with Unreal Engine as Blueprints or C++ are required when not using the template projects. In Unity, this simply involves inserting a character controller included with the engine into the scene.

Animating cameras and objects in the scene is accomplished in both engines by creating keyframes in an animation panel. However, Unreal Engine does have the functionality built-in to save videos of game output, while in Unity this must be programmed or found in the Unity Asset Store.

For simulations such as crowd movement, both engines can build navigation meshes using imported geometry to create a surface for characters to navigate. The main difference is how the agents' destinations are programmed. In Unity, this involves a C# script while Unreal requires Blueprints or C++.

Pricing

Unity

The chart below outlines the differences between the versions of Unity.

| | Unity Personal | Unity Plus | Unity Pro |
|-----------------------------|----------------------------------|----------------------------------|---------------------|
| Cost | \$0 | \$35 per month | \$125 per month |
| Asset Store Discount | | 20% off asset store | 20% off asset store |
| Splash Screen Customization | No | Yes | Yes |
| Source Code Access | No | No | Yes |
| Tier Eligibility | Company revenue less than \$100k | Company revenue less than \$200k | |

Based on the pricing model, Unity Personal is the best choice for projects where a splash screen is acceptable and company revenue is less than \$100 000. Unity Plus or Unity Pro are required as company revenue increases.ⁱⁱⁱ

Unreal

Unreal Engine charges royalties for the work created with the engine. There is a 5% royalty on the product revenue after the first \$3000 per game per calendar year^{iv} However, videos created with Unreal Engine that generate revenue do not fall under the royalty clause.^v

Unreal Studio has a royalty free pricing model, with a cost of \$49 per month when purchased with an annual subscription.

Comparison

For creating interactive simulations, Unreal Engine can be used for free when using the FBX import workflow provided no revenue is generated from the created simulation. However, because the CityEngine to Unreal FBX workflow is not recommended due of the performance benefits of using Datasmith, comparing Unreal Studio to Unity is more useful. The \$49 a month cost of Unreal Studio is competitive as it falls between the free and \$125 monthly costs of the different versions of Unity.

Conclusion

Unity and Unreal Engine are very useful tools for visualizations and simulations of GIS data. Experimenting with both engines before purchasing a license is recommended as they can be downloaded for free to evaluate.

Unity is the recommended starting point for a beginner as there are less hurdles to create an interactive simulation of a 3D environment than Unreal Engine. The Unity Asset Store also has a more content available than the Unreal Marketplace, meaning a beginner is more likely to find a required asset than need to model or develop it themselves.

Unreal Studio, which is scheduled to be released from beta at the end of 2018, should be considered for projects with longer time frames that allow more flexibility at the project start to learn the engine. In return for the steeper learning curve, the tighter integration with CityEngine may increase long term productivity by reducing the import time for large instanced 3D models.

ⁱ <https://docs.unity3d.com/Manual/3D-formats.html>

ⁱⁱ <https://docs.unrealengine.com/en-us/Engine/Content/FBX>

ⁱⁱⁱ <https://unity3d.com/legal/terms-of-service/software>

^{iv} <https://www.unrealengine.com/en-US/faq?active=general>

^v <https://www.unrealengine.com/en-US/faq?active=legal>