Cubiquity for Unity3D Documentation

Congratulations! If you are reading this user manual then you have discovered Cubiquity for Unity3D, and are about to take a step into the exciting world of voxels. Over the coming pages we will show you how voxel-based worlds can enhance your games or virtual environments, and make sure you have the knowledge required to use Cubiquity for Unity3D to its maximum potential. You can start by reading one of the sections listed below.

Installation

You can obtain Cubiquity for Unity3D from the asset store or from our Git repository. Or perhaps you have it already and are reading a local copy of this documentation. In any case you will want to add it to your project and this section explains how. **Read more...**

Quick Start

We can't blame you for wanting to get started quickly, and often the best way to learn is by experimentation. This section will just give you the basic knowledge you need to create your first voxel environments. You can then read other sections of this manual when you want more details of a specific topic. **Read more...**

Main Principles

To make the most of Cubiquity for Unity3D you will want an understanding of the different components of the system and how they fit together. This section provides a discussion of the key concepts behind Cubiquity for Unity3D and presents the different classes you will be working with. A good understanding of this section will be important to make effective use of our system, particularly if you want to make a lot of use of it from code. **Read more...**

User Interface

The easiest way to get started making voxel environments is to use the built-in editing tools. These are embedded within the Unity3D editor, and they let you directly manipulate your worlds though the scene view. You can create worlds, edit them, and set rendering or physical properties. In this section of the user manual we take you though all the tools which are at your disposal and give tips for getting the most out of them. **Read more...**

Obtaining Volume Data

You may wish to generate or obtain volume data from external sources and then bring it into Cubiquity. In this section we explain some of your options for doing so. **Read more...**

Trouble Shooting

Sometimes things can go wrong, and if they do then this is the place to look for help. **Read more...**

Class List

Cubiquity for Unity3D provides a clean and elegant API for working with volumes. You can use this API to create volumes procedurally or import them from external sources, modify your volumes in response to in-game events (such as explosions), or implement custom editing tools for your players to use. This section describes the classes and method which you can use for this purpose **Read more...**