

# CRYENGINE<sup>®</sup>

## CRYENGINE 5.3 - Getting Started Guide

---

**Version 1.0.1**

# Getting Started With CRYENGINE Guide

---

## Table of Contents

<b>CHAPTER 1: INTRODUCTION.....</b>	<b>4</b>
<b>CHAPTER 2: SYSTEM REQUIREMENTS.....</b>	<b>5</b>
<b>CHAPTER 3: MANAGING CRYENGINE.....</b>	<b>6</b>
CREATING A CRYENGINE ACCOUNT .....	6
DOWNLOADING/INSTALLING THE CRYENGINE LAUNCHER.....	8
CRYENGINE INSTALLATION .....	10
Creating a New Project - No Engine Installed .....	10
UPGRADING CRYENGINE .....	14
<b>CHAPTER 4: MANAGING CRYENGINE PROJECTS .....</b>	<b>19</b>
CREATING A NEW PROJECT - ENGINE VERSION ALREADY INSTALLED .....	19
ADDING ASSETS TO YOUR PROJECT .....	23
IMPORTING AN EXISTING PROJECT .....	27
<b>CHAPTER 5: INSTALLING CRYENGINE PLUGINS AND TOOLS .....</b>	<b>30</b>
CRYTOOLSINSTALLER .....	30
RUNNING THE CRYTOOLSINSTALLER .....	30
SELECTING PACKAGES TO INSTALL .....	31
PROJECT LAUNCHER TOOLS .....	33
<b>CHAPTER 6: SANDBOX EDITOR OVERVIEW .....</b>	<b>34</b>
CREATING A NEW LEVEL .....	34
CRYENGINE OBJECTS .....	35
EDITOR LAYOUT.....	36
ESSENTIAL TOOLS EDITORS.....	36
PLACING AN OBJECT IN YOUR LEVEL .....	36
TRANSFORMING OBJECTS .....	37
Select.....	37
Move .....	39
Rotate.....	40
Scale .....	41
Coordinates System.....	42
Snap Feature .....	44
BASIC NAVIGATION - KEYBOARD SHORTCUTS IN CRYENGINE .....	45
Game Mode.....	45
<b>CHAPTER 7: EDITOR TOOLS .....</b>	<b>46</b>
MATERIALS.....	46
Creating a New Material.....	46

---

# Getting Started With CRYENGINE Guide

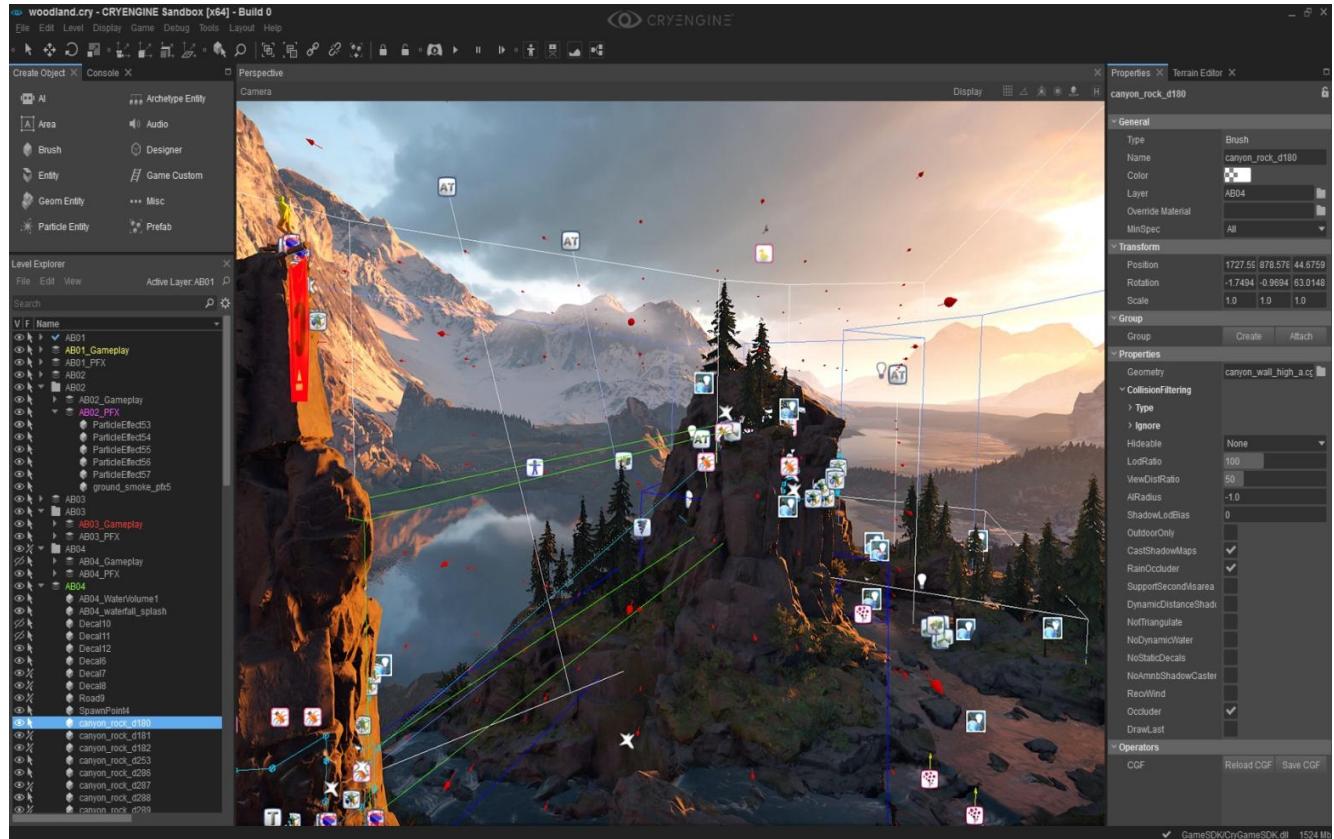
---

Creating a New Multi-Material.....	48
Assigning a Material to an Object.....	49
<b>BRUSHES.....</b>	<b>53</b>
<b>TERRAIN .....</b>	<b>55</b>
Generating a New Terrain .....	55
Sculpting a New Terrain.....	57
Painting a New Terrain .....	57
<b>ADDING SKY TO YOUR LEVEL .....</b>	<b>59</b>
<b>LIGHTING .....</b>	<b>60</b>
Time of Day .....	60
Environment Probes.....	60
Lights .....	61
<b>PLACING A CAMERA.....</b>	<b>63</b>
<b>PHYSICS.....</b>	<b>64</b>
How to use .....	65
Controls .....	65
Technical information.....	65
CVars .....	66
<b>PHYSICS ENTITIES.....</b>	<b>66</b>
<b>VISUAL SCRIPTING.....</b>	<b>67</b>
Flow Graph Scripting .....	67
Debug Flow Graph.....	73
<b>SCHEMATYC.....</b>	<b>74</b>
Basic Concepts.....	74
<b>PARTICLE SYSTEM .....</b>	<b>76</b>
Creating and Adding a Particle Effect in Your Level .....	76
<b>CHAPTER 8: USING GITHUB FOR CRYENGINE.....</b>	<b>79</b>
CONFIGURING A GIT CLIENT .....	79
UPDATING YOUR REPOSITORY .....	80
RESTRICTED ACCESS SOURCE CODE .....	81
<b>CHAPTER 9: GLOSSARY/TERMINOLOGY.....</b>	<b>82</b>

## Chapter 1: Introduction

[CRYENGINE](#) is one of the most powerful game engines in the industry and allows teams of all sizes around the world to create world-class entertainment experiences with the latest game development technologies.

For more information about sample creation projects, please refer to the [Tutorials](#) section of our documentation.



## Chapter 2: System Requirements

Your PC **MUST** meet at least the minimum specification before you install CRYENGINE.

Minimum	Recommended
<ul style="list-style-type: none"><li>• <b>OS (to run Game):</b> Windows 7, 8.1, 10 (32-bit or 64-bit)</li><li>• <b>OS (to run Editor):</b> Windows 7, 8.1, 10 (64-bit only)</li><li>• <b>Processor:</b> Intel Dual-Core min 2GHz (Core 2 Duo and above) or AMD Dual-Core min 2GHz (Phenom II X2 and above)</li><li>• <b>Memory:</b> 4 GB RAM</li><li>• <b>Graphics:</b> NVIDIA GeForce 450 series or AMD Radeon HD 5750 series or higher (minimum 1 GB dedicated VRAM GDDR5)</li><li>• <b>DirectX:</b> Version 11</li><li>• <b>Hard Drive:</b> 8 GB available space</li><li>• <b>Sound Card:</b> DirectX Compatible Sound Card with latest drivers</li></ul>	<ul style="list-style-type: none"><li>• <b>OS (to run Game):</b> Windows 7, 8.1, 10 (64-bit)</li><li>• <b>OS (to run Editor):</b> Windows 7, 8.1, 10 (64-bit)</li><li>• <b>Processor:</b> Intel Quad-Core (i5 2300) or AMD Octo-Core (FX 8150)</li><li>• <b>Memory:</b> 8 GB RAM</li><li>• <b>Graphics:</b> NVIDIA GeForce 660Ti or higher, AMD Radeon HD 7950 or higher (minimum 2 GB dedicated VRAM GDDR5)</li><li>• <b>DirectX:</b> Version 11</li><li>• <b>Hard Drive:</b> 8 GB available space</li><li>• <b>Sound Card:</b> DirectX Compatible Sound Card with latest drivers</li></ul>

# Getting Started With CRYENGINE Guide

---

## Windows Vista SP1

As of CRYENGINE version 5.3.0 Windows Vista SP1 is no longer supported.

## Video and Sound Card Drivers

Ensure that you have the most up to date video and sound card drivers installed on your PC.

## Integrated GPUs and DirectX 11

CRYENGINE might run on the integrated GPUs from Intel and AMD (3<sup>rd</sup> Generation Intel i-core CPUs and above, AMD APUs that support DX11) but they are not officially supported. We advise you to not run CRYENGINE on integrated Graphics solutions due to sub-optimal performance.

Please also keep in mind that integrated GPUs share RAM with the CPU, which results in less available RAM for the CPU and OS. Hence, 4GB of RAM might not be enough and therefore minimum System Requirements will not be met.

Furthermore, for full DirectX 11 compatibility not only does the OS have to support it, but so does the GPU and at hardware feature level. Therefore, it doesn't matter that the Windows OS is DirectX 11 capable, the GPU MUST be too.

Please check your GPU manufacturer's website for more info - [Nvidia](#) or [AMD](#)

## Software

Ensure that you have the most [up to date version of Direct X](#) installed on your PC – this is particularly important after installing Windows from scratch when it is advisable to re-run the installation manually.

It is also advisable to install Visual C++ redistributables. These are available [here](#). This is particularly important if you have performed a new installation of Windows (minimum version Visual C++ 2015).

.NET Framework should be installed and up to date. It is available [here](#).

## Windows Update

If you encounter issues with API-MS-WIN-CRT-RUNTIME-L1-1-0.DLL or similar missing on launch, then please ensure that Windows Update is enabled. Specifically, this [update](#) is required to be installed

## Chapter 3: Managing CRYENGINE

To install CRYENGINE you will need to create a CRYENGINE account and then download the CRYENGINE Launcher. Once installed, the Launcher allows you to install and manage CRYENGINE and create/edit your projects. To create a CRYENGINE account, go to [cryengine.com](#) and perform the steps below.

- [Creating a CRYENGINE Account](#)
- [Downloading the CRYENGINE Launcher](#)
- [CRYENGINE Installation](#)
- [Upgrading CRYENGINE](#)

### Creating a CRYENGINE Account

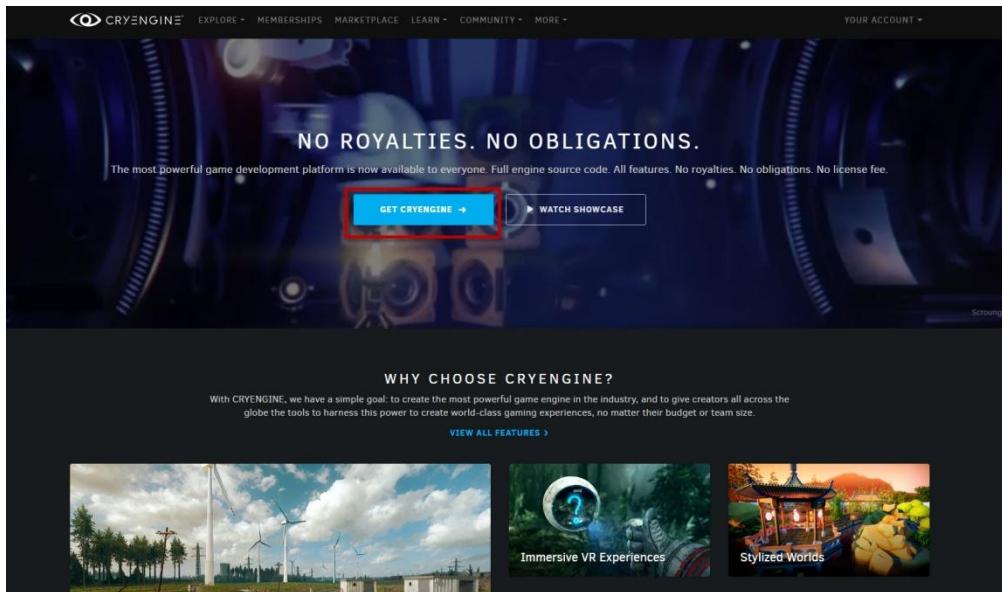
To install and manage CRYENGINE, you need a CRYENGINE account. Your CRYENGINE account allows you to access the CRYENGINE community, the CRYENGINE marketplace, and learn more about CRYENGINE.

**Note:** If you already have a CRYENGINE account you can log in using your existing credentials to proceed with CRYENGINE installation.

# Getting Started With CRYENGINE Guide

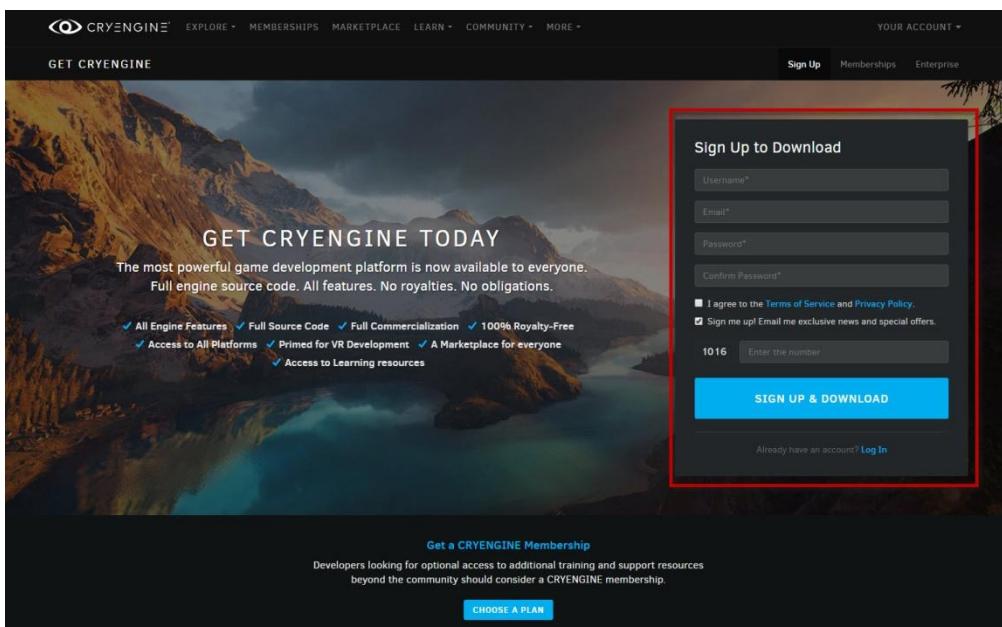
You can create a CRYENGINE account in minutes. Follow the simple steps below.

1. Go to [cryengine.com](https://cryengine.com) and then click the **GET CRYENGINE** button.



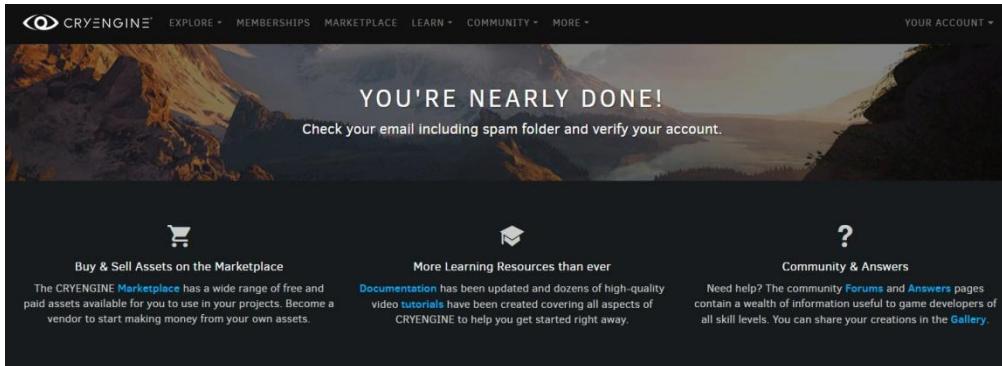
2. Enter your details in the **Sign Up to Download** section, and then click the **Sign Up & Download** button.

**Note:** Make sure that you have read and checked CRYENGINE's Terms of Service and Privacy Policy before you click the **Sign Up & Download** button.



3. Check your email account and verify your newly created CRYENGINE account.

# Getting Started With CRYENGINE Guide



4. Congratulations, you have successfully created a CRYENGINE account!

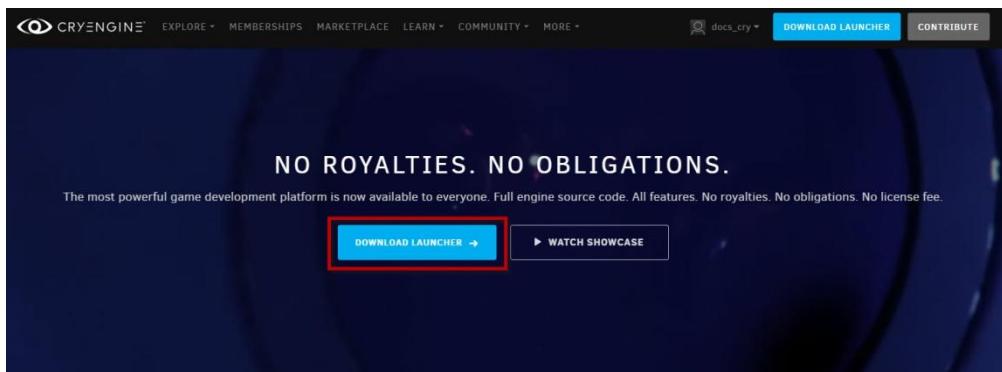
## Downloading/Installing the CRYENGINE Launcher

Don't forget that you need a CRYENGINE account before you can install the CRYENGINE Launcher. Need an account? Then go to [cryengine.com](https://cryengine.com) and click the **GET CRYENGINE** button.

After you have successfully created a CRYENGINE account, download the CRYENGINE Launcher by signing into your CRYENGINE account with your newly created credentials.

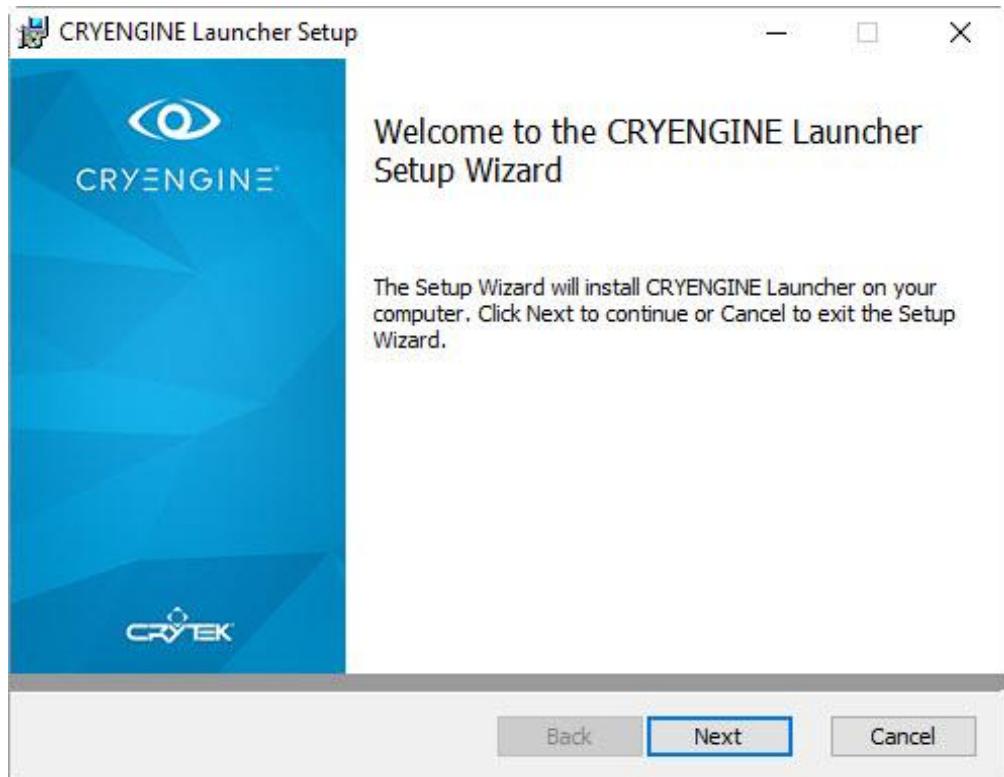
Download the CRYENGINE Launcher by following the steps below.

1. Log in into your CRYENGINE account.
2. From the webpage, click the **Download Launcher** button to initialize the download.

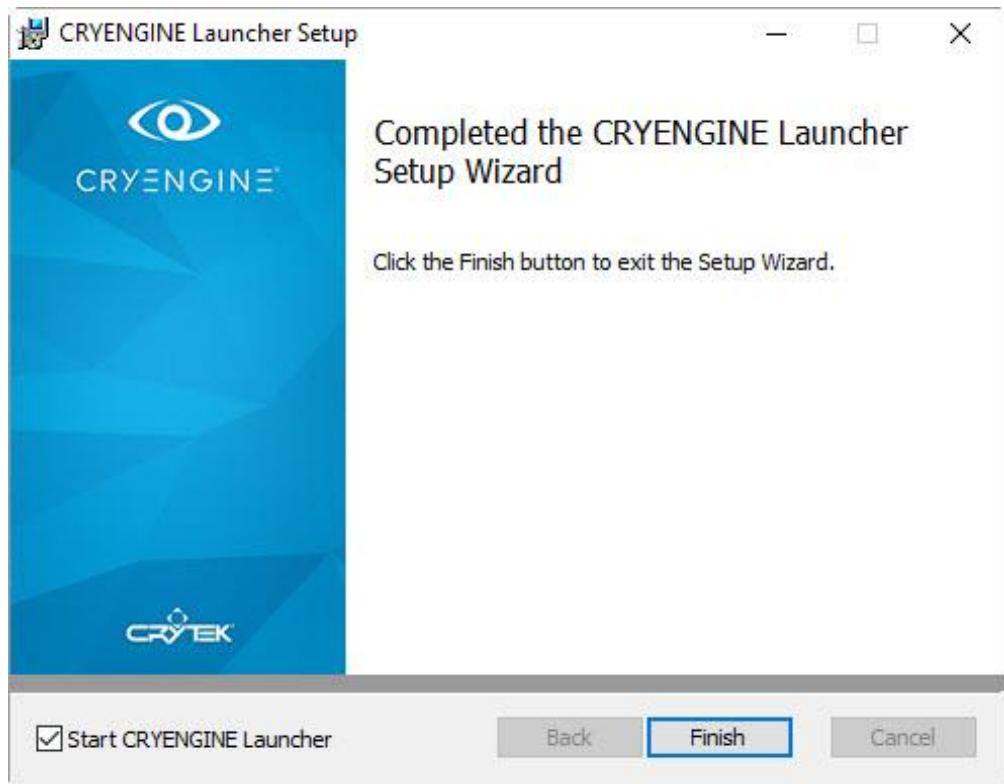


3. Open the downloaded file, and follow the onscreen instructions to complete the installation.

## Getting Started With CRYENGINE Guide



4. Click the **Finish** button to complete the installation and launch the CRYENGINE Launcher.



5. Congratulations, you have successfully completed the Launcher installation.

## CRYENGINE Installation

CRYENGINE can only be installed through the CRYENGINE Launcher, which requires a user to be logged-in to their CRYENGINE account.

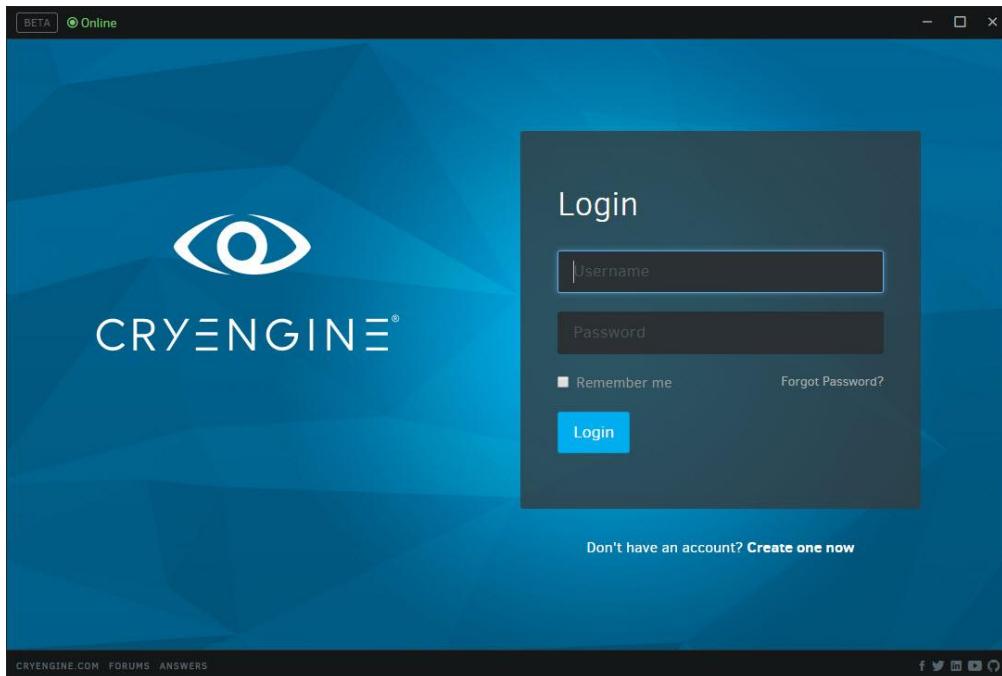
A project must always be created/saved first and only then can an Engine version be downloaded/installed. The CRYENGINE Launcher manages this process, but it's important to understand the workflow.

**Note:** You can download/install more than one version of the Engine by creating new projects for each Engine version.

### ***Creating a New Project - No Engine Installed***

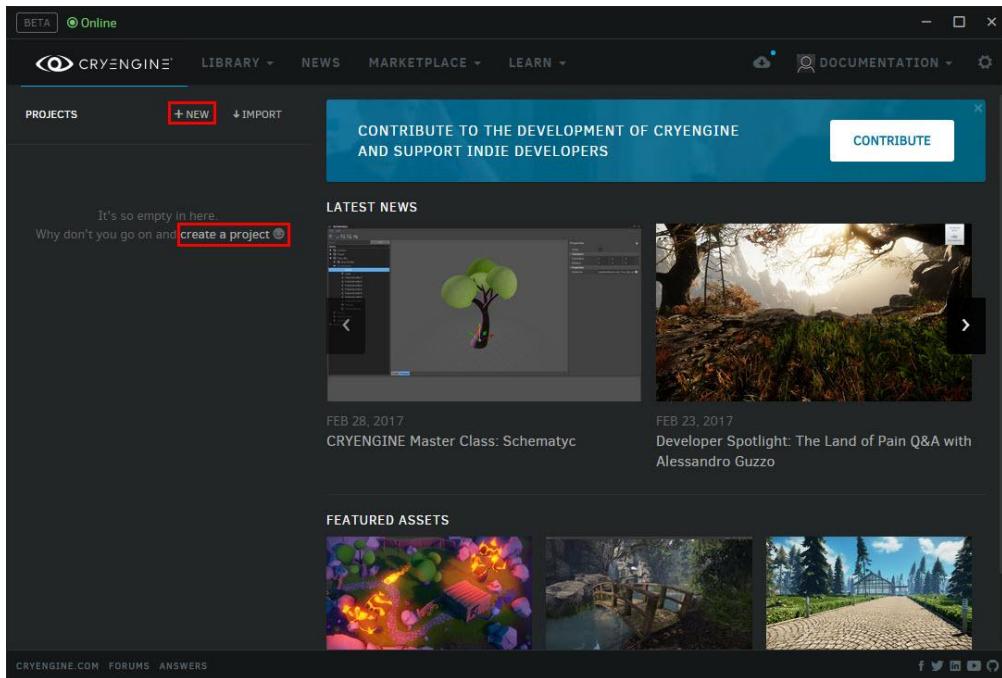
Follow the steps below to create a new CRYENGINE project using the Launcher.

1. Open the CRYENGINE Launcher from the icon on your desktop and then log in to your account.

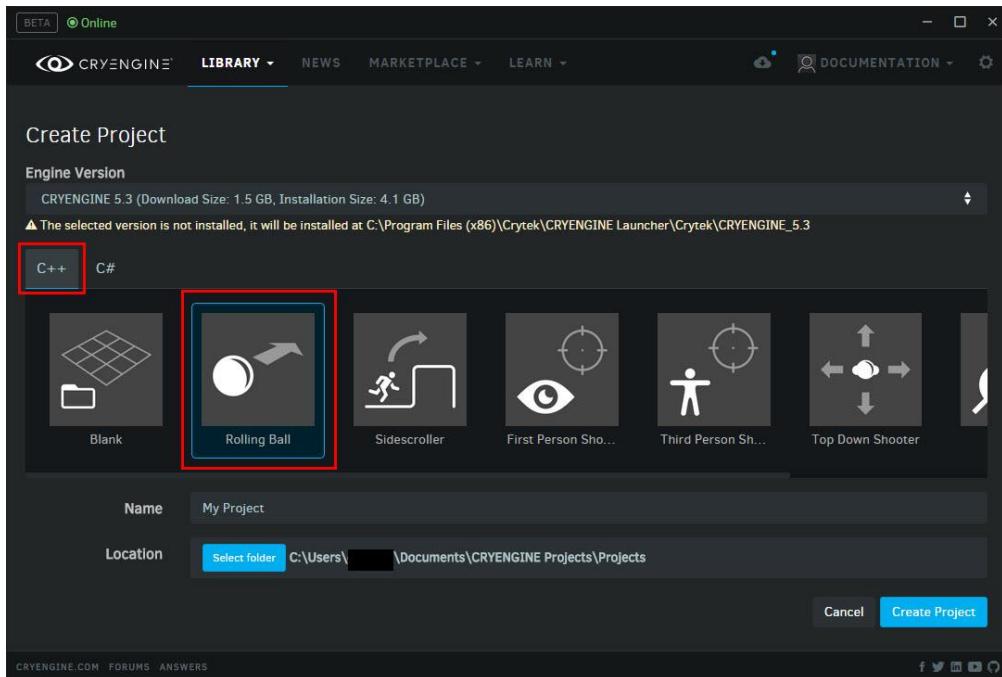


2. In the Launcher main window, click on either the **+ New** or the **create a project** option.

# Getting Started With CRYENGINE Guide

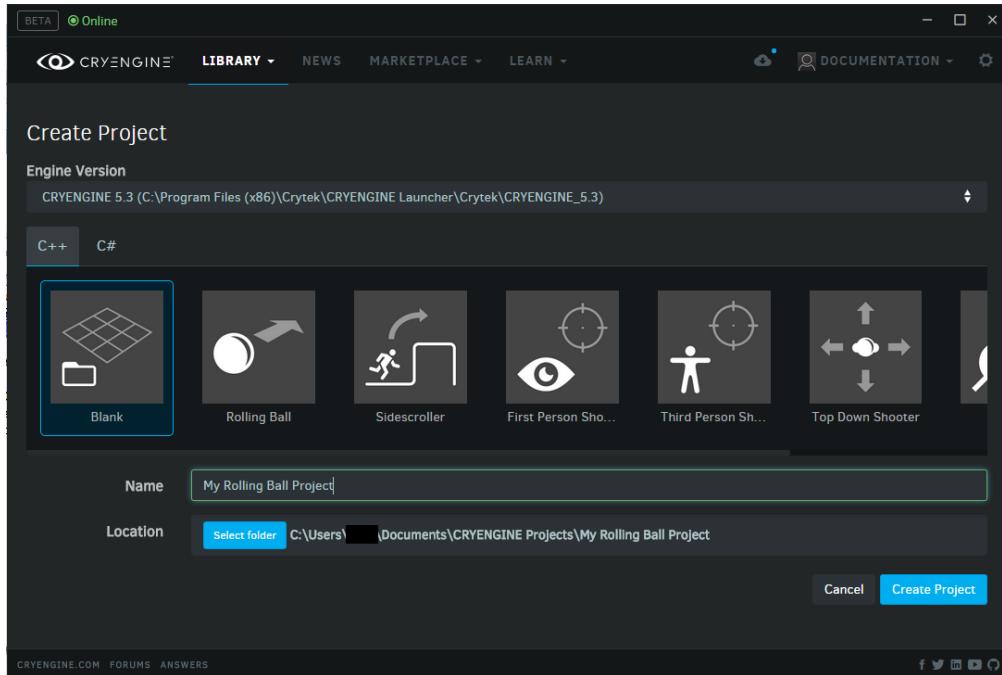


3. You can create a project based on C++ or C#. There are a number of ready-made templates that make things much easier for you when getting started with CRYENGINE. In the example below, we have chosen a C++ based project using the Rolling Ball template. Remember that in order to be able to install an Engine version you need to have a project in the Launcher!

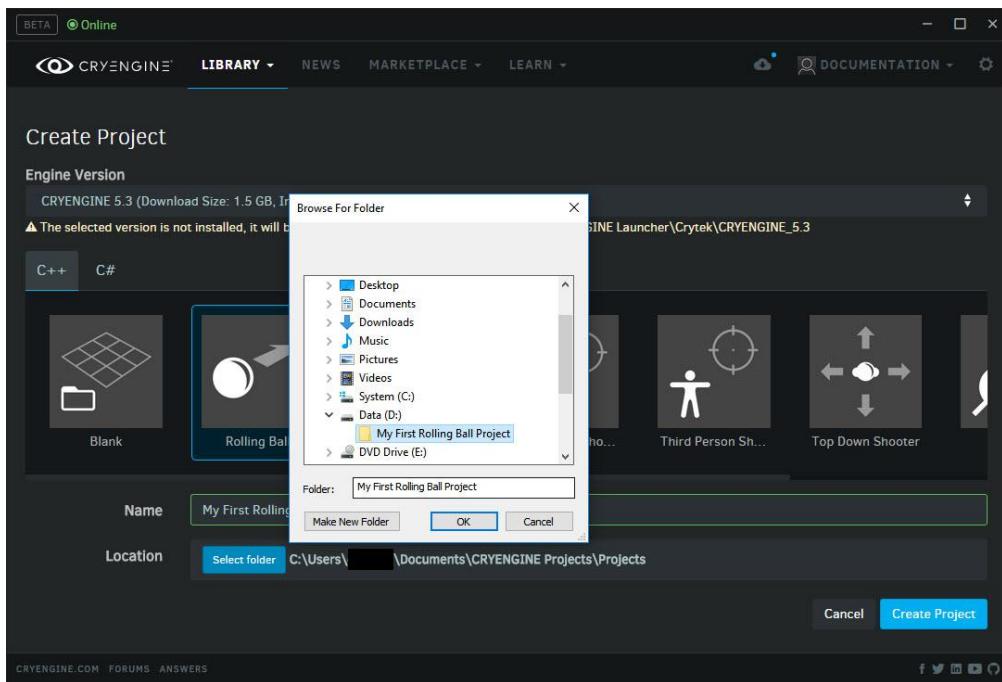


# Getting Started With CRYENGINE Guide

4. Specify a name for your project. In the example below, we have named our project *My First Rolling Ball Project*.

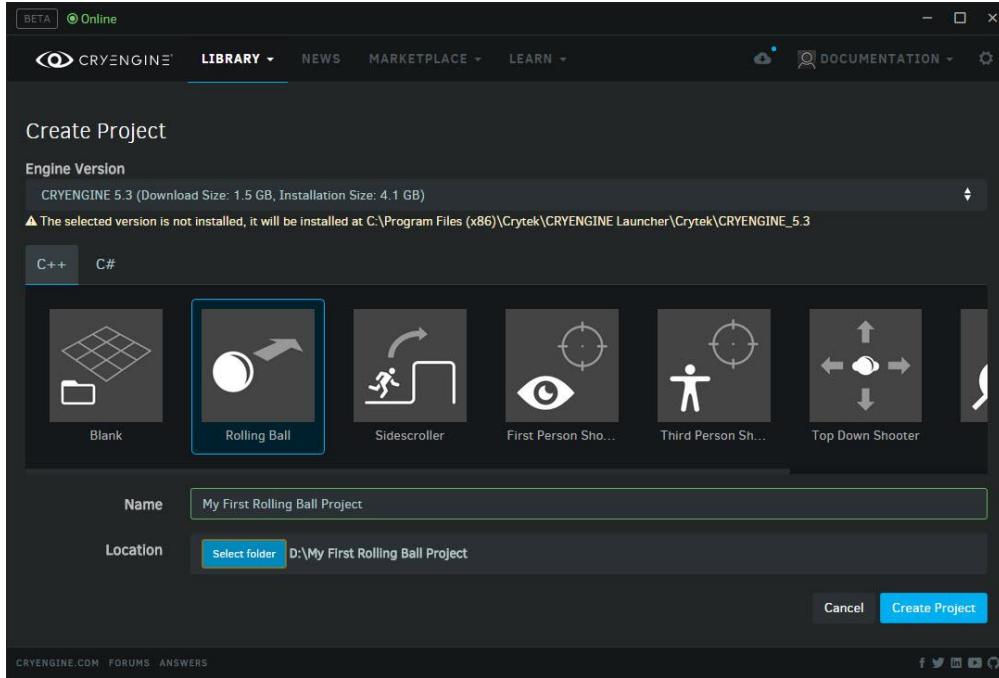


5. By default, Launcher saves you projects in a predefined location. You can modify the Engine install location by selecting the **Select Folder** button; this will open the **Browse For Folder** pop-up. Next, navigate to the location where you want to save your project (in the example, we have used the D drive), and then click **OK**.

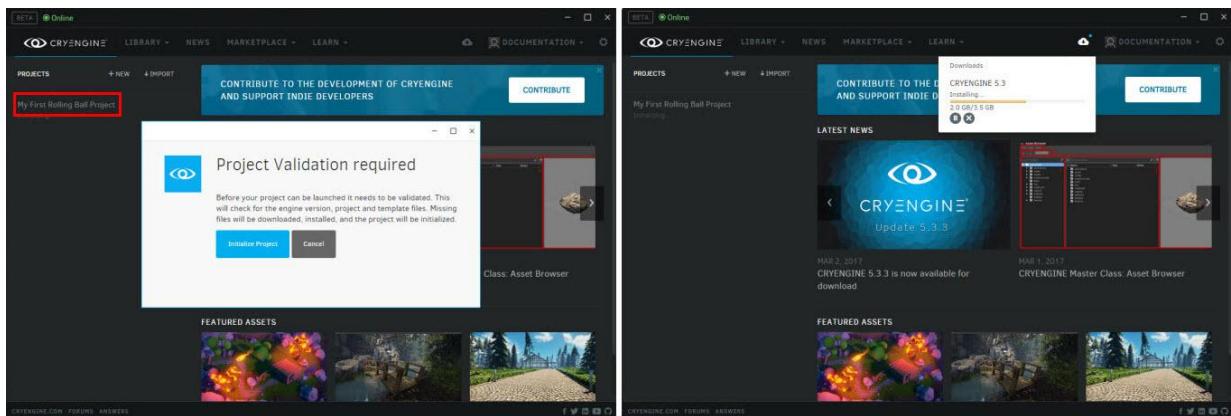


# Getting Started With CRYENGINE Guide

- The Location will now change accordingly and you can create your project by clicking the **Create Project** button.

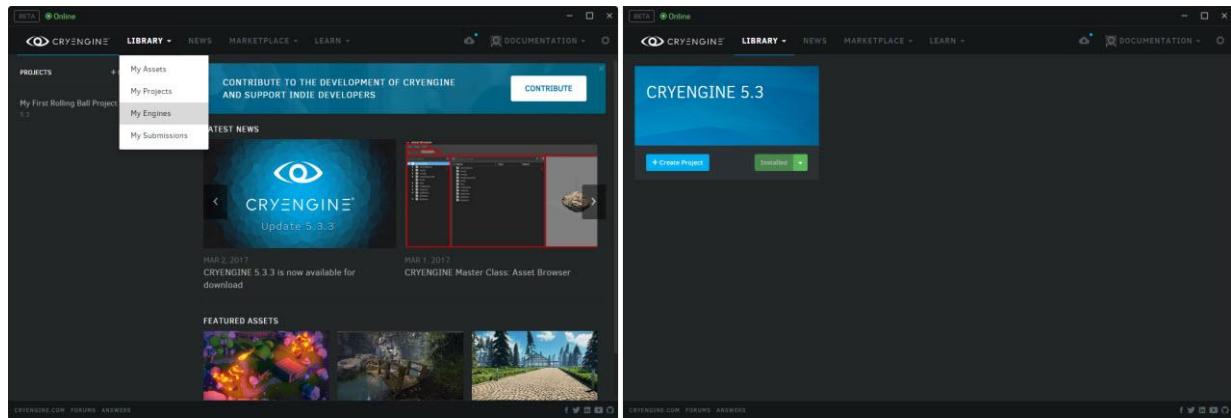


- Your project will now appear in the list of Projects, but as no Engine version is currently installed you need to initialize your project. Click on the **Initialize Project** button which will start the Engine download/install process.



- Once the download/installation process is complete, you can confirm the Engine version that has been installed by hovering over the **Library** menu option and then clicking on **My Engines**. In our example CRYENGINE 5.3 has been installed.

# Getting Started With CRYENGINE Guide

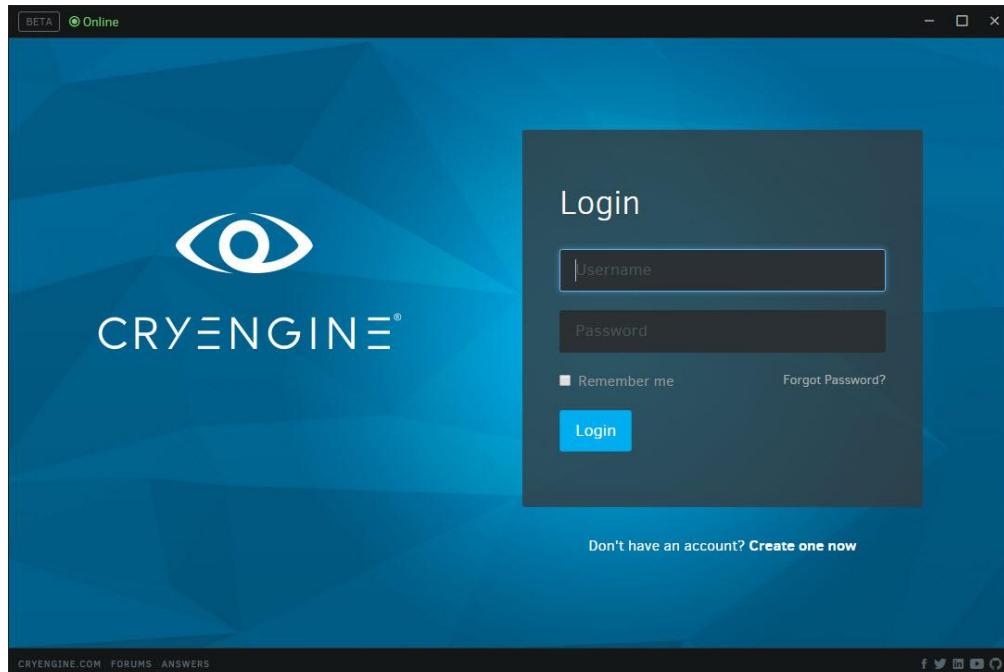


## Upgrading CRYENGINE

The latest version of CRYENGINE can be downloaded through the Launcher. Follow the steps below to update an older version of the Engine to the latest available version.

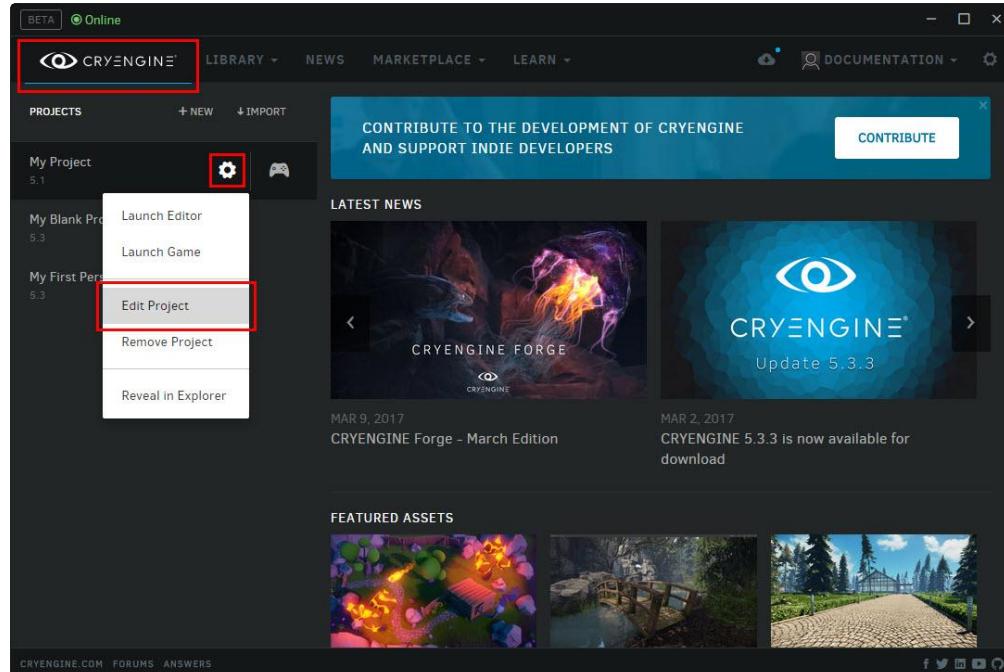
**Note:** To upgrade from CRYENGINE 5.2 to 5.3 see the article [here](#).

1. Open the CRYENGINE Launcher from the icon on your desktop and then log in to your account.



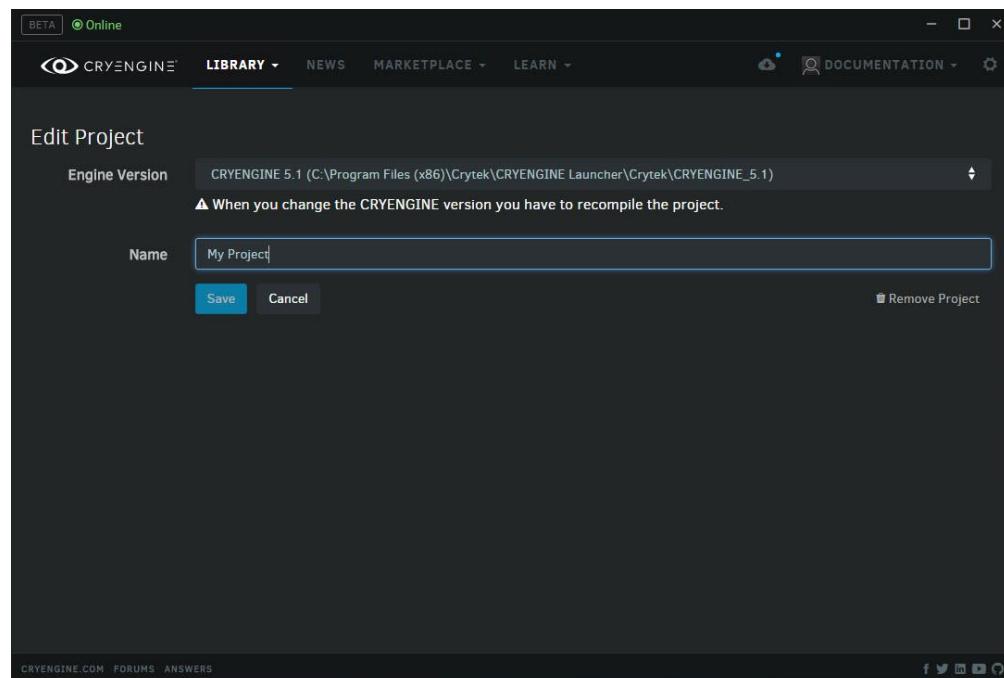
# Getting Started With CRYENGINE Guide

2. In the main Launcher window, hover over the project which you want to update and then click the **Gear** icon to select the **Edit Project** option. In the example below, we are going to upgrade the Engine in the project named My Project (currently using Engine version 5.1).



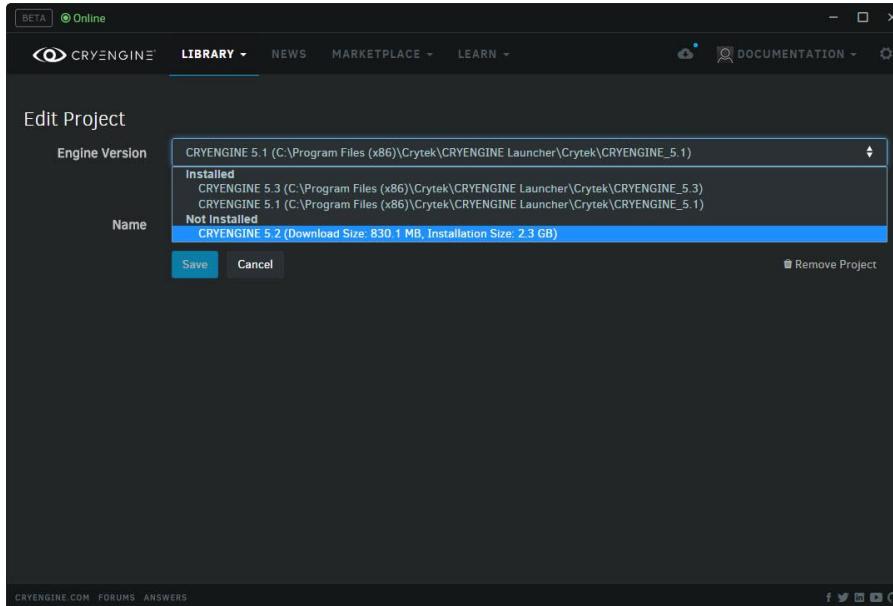
3. Clicking on **Edit Project** opens the Edit Project window where the currently installed Engine Version is shown (for My Project).

**Note:** Upgrading to a newer version of the Engine requires you to recompile the project (see step 6 below).

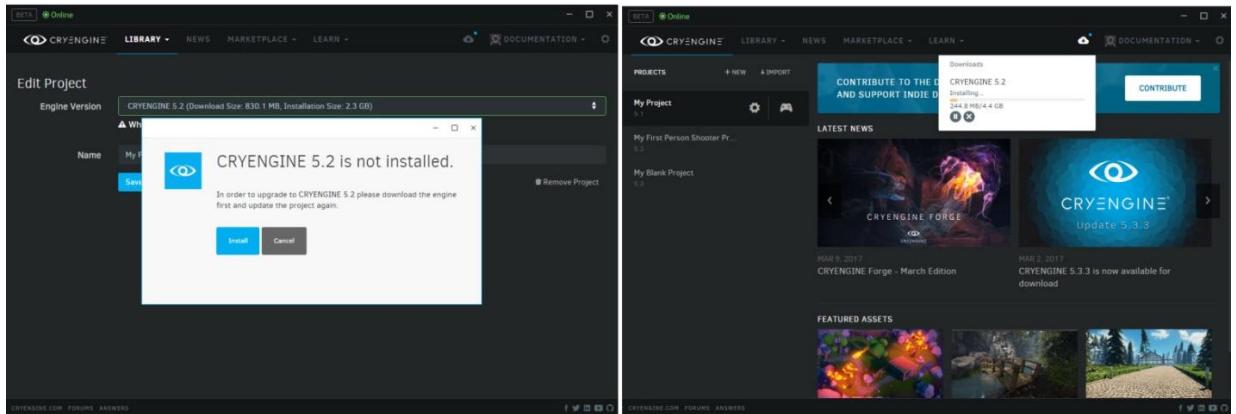


# Getting Started With CRYENGINE Guide

4. Select the **Engine Version** drop-down menu to reveal the Engine versions that you have installed and Engine versions that are available (but uninstalled). In the example below, we have selected CRYENGINE 5.2. Then click the **Save** button.

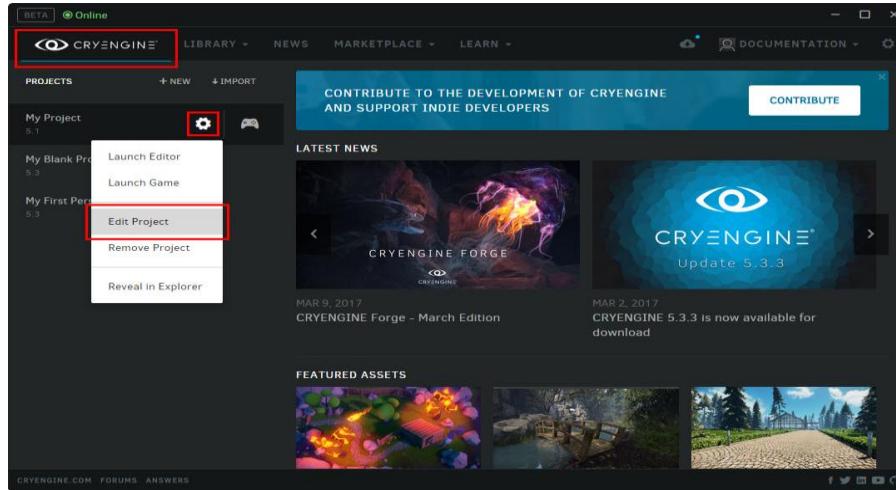


5. You now need to download and install CRYENGINE 5.2 by clicking the **Install** button.

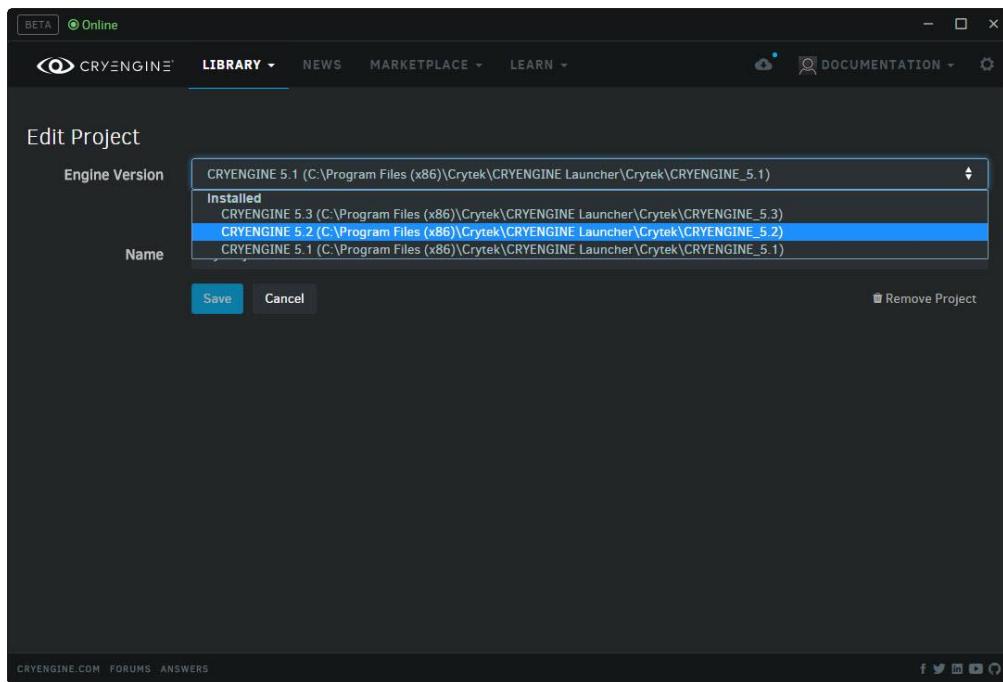


# Getting Started With CRYENGINE Guide

6. The next step is to upgrade the project (recompile). This is a repeat of step 2 above.



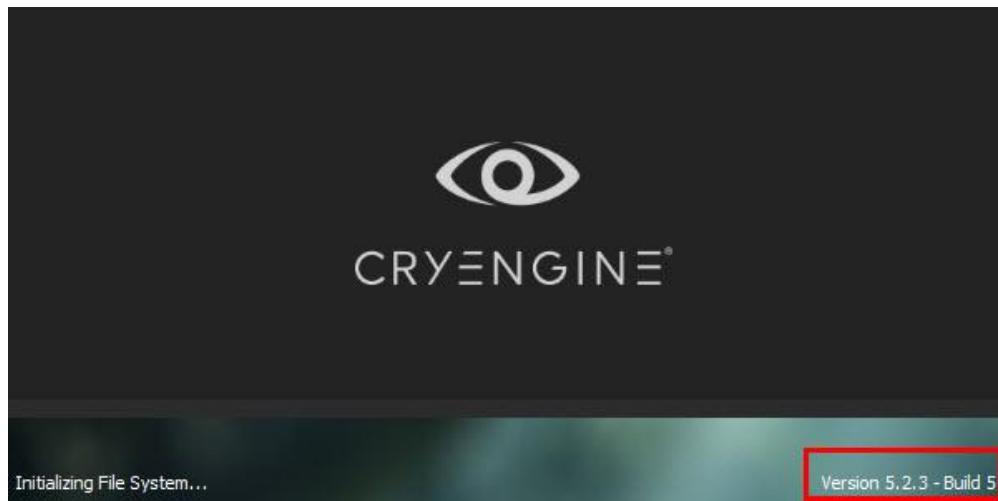
7. The Edit Project window will open. Use the Engine Version drop-down menu to select the Engine version you are upgrading to, in our example CRYENGINE 5.2. Then click the **Save** button.



## Getting Started With CRYENGINE Guide

---

8. You can confirm that your project has been upgraded by Launching the Editor and checking the Engine version being used.



# Chapter 4: Managing CRYENGINE Projects

The CRYENGINE Launcher allows you to manage all of your CRYENGINE projects in one location. This includes project creation, editing, and removal.

**Note:** On initial CRYENGINE installation (no Engine previously installed) the most up to date version of the Engine at the time of download/installation will be installed.

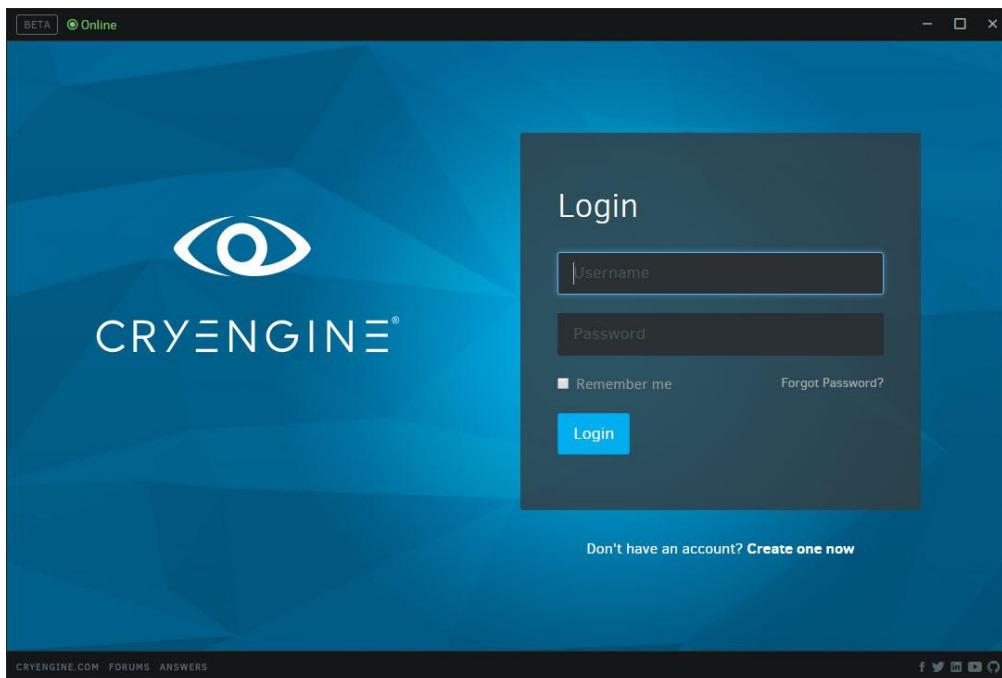
It is possible for more than one version of the Engine to be downloaded/installed, but the requirement that every Engine version must first have a project created/installed remains.

## Creating a New Project - Engine Version Already Installed

- [Adding Assets to your Project](#)
- [Import an Existing Project](#)

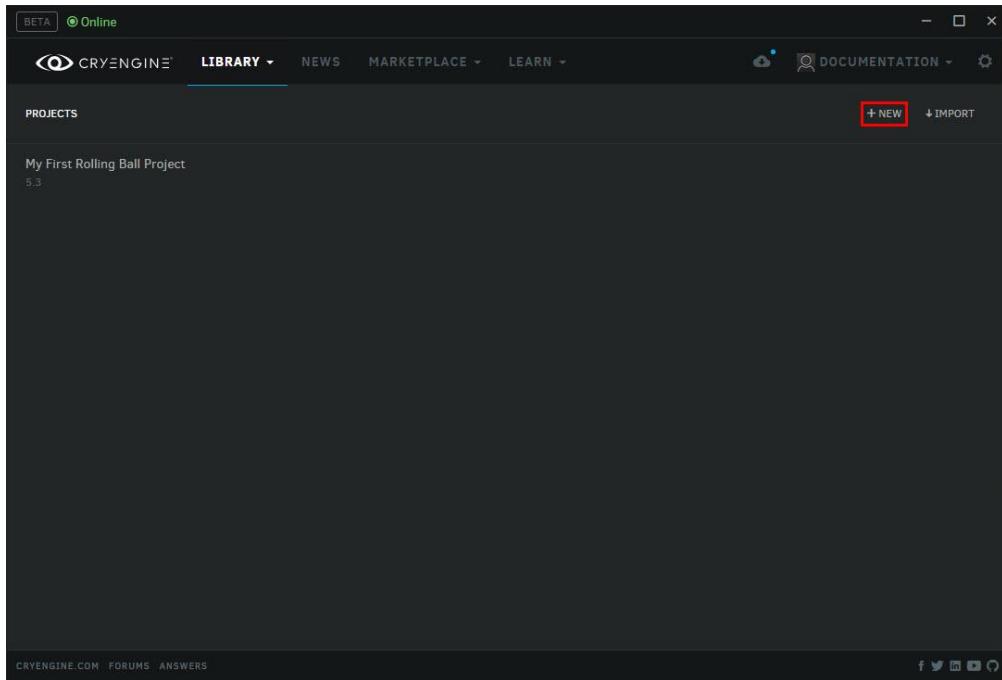
Follow the steps below to create a new CRYENGINE project using the Launcher.

1. Open the CRYENGINE Launcher from the icon on your desktop and then log in to your account.

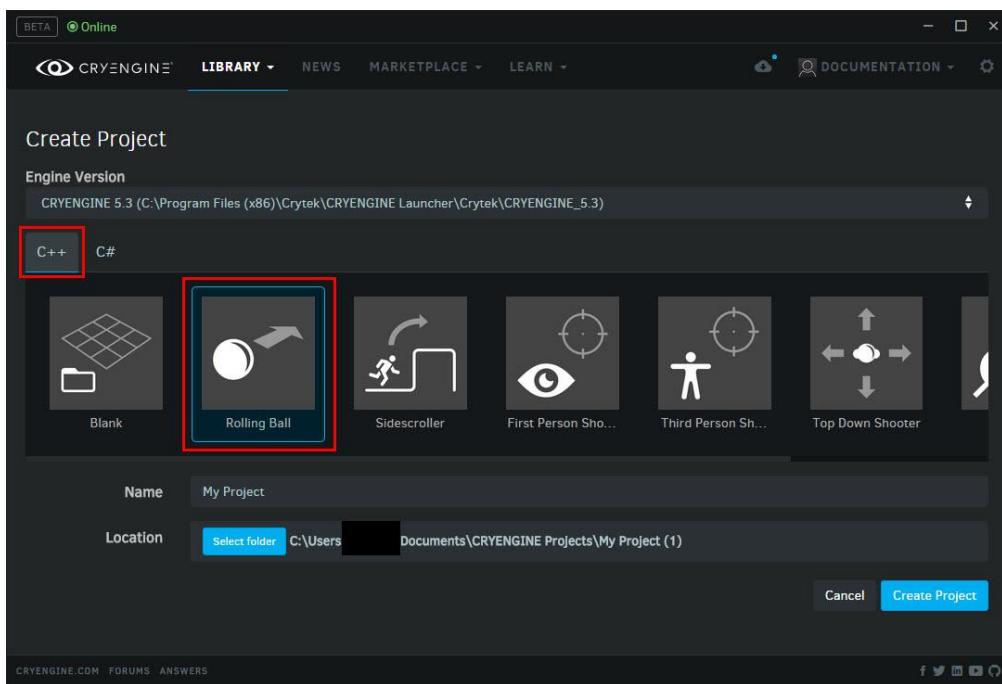


# Getting Started With CRYENGINE Guide

2. In the Launcher main window, under the **LIBRARY** menu **My Projects**, click the **+ New** option.

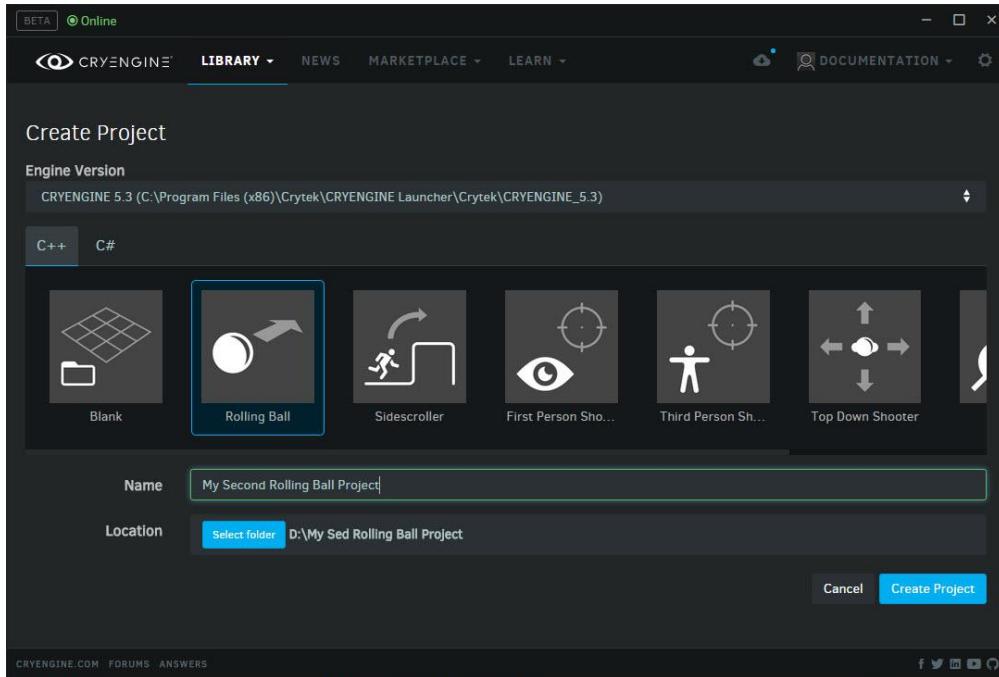


3. You can create a project based on C++ or C# programming knowledge. There are a number of ready-made templates that make things much easier for you when getting started with CRYENGINE. In the example below, we have chosen a C++ based project using the Rolling Ball template and have chosen Engine version 5.3. Remember that in order to install an Engine version, you need to have a project in the Launcher!

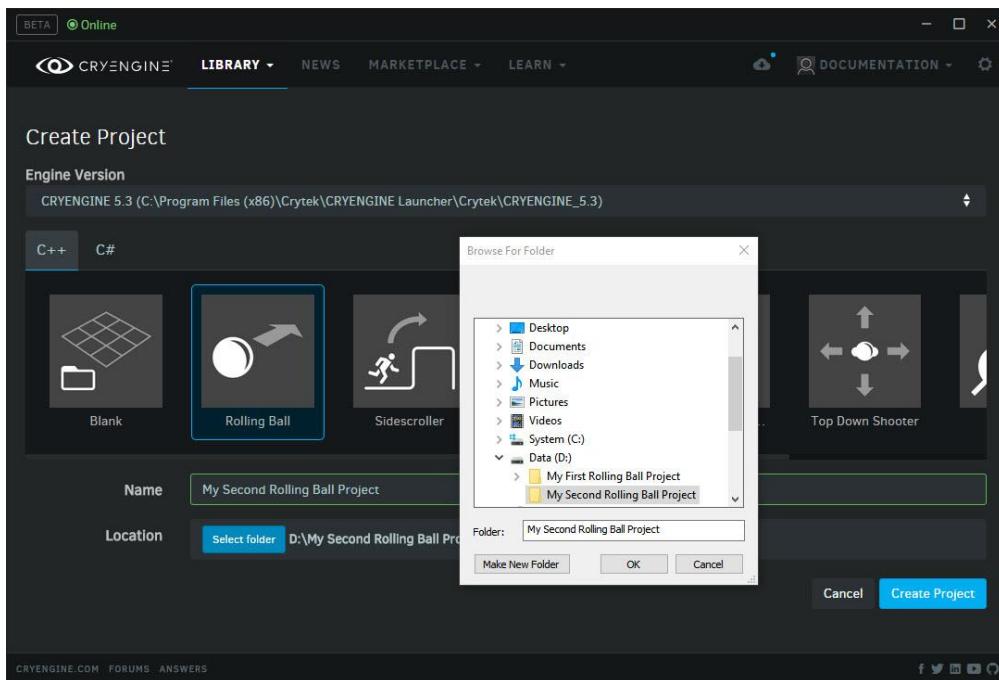


# Getting Started With CRYENGINE Guide

- Now choose a name for your project. In the example below, we have named our project "My Second Rolling Ball Project."

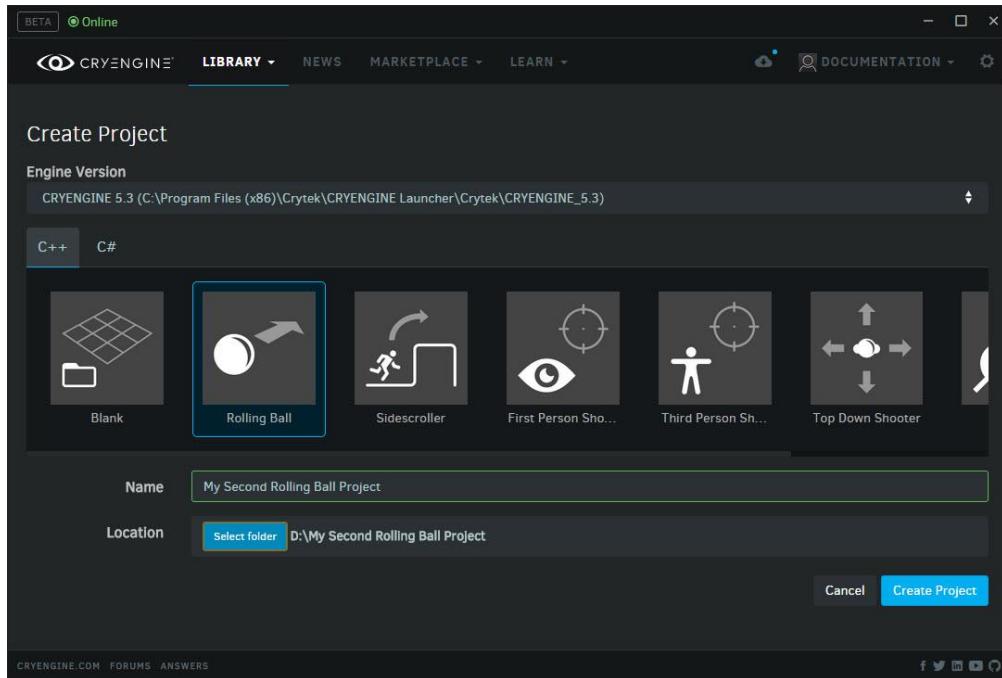


- By default, the Launcher saves you projects in a predefined location. You can modify the Engine install location by clicking the **Select Folder** button; this will open the **Browse For Folder** pop-up. Next, navigate to the location where you want to save your project (in the example, we have used the D drive), and then click **OK**.

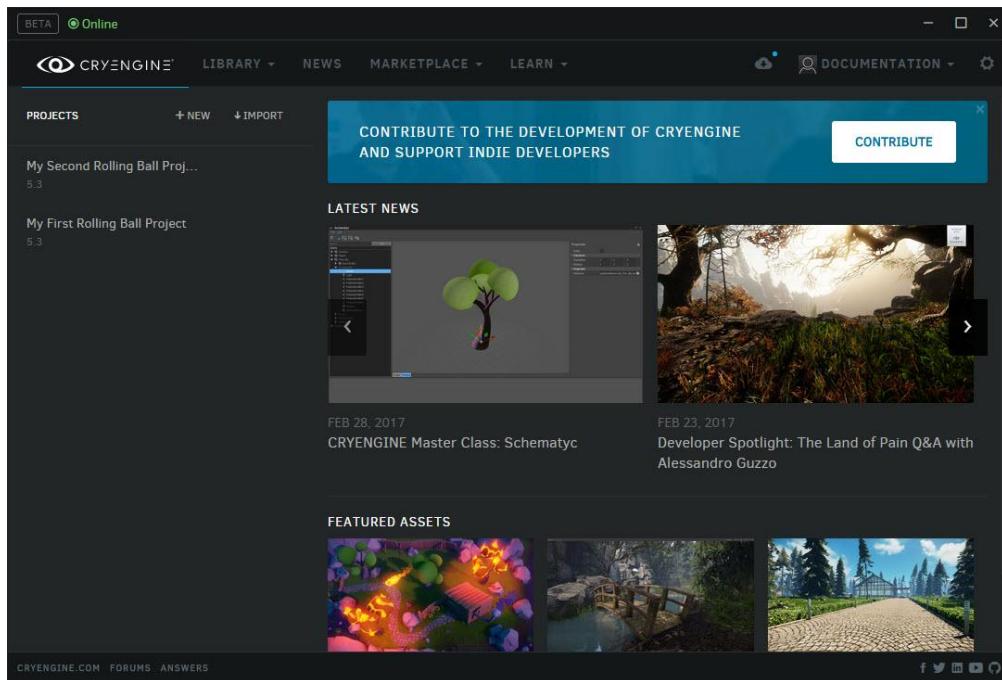


# Getting Started With CRYENGINE Guide

6. The Location will now change accordingly and you can create your project by clicking **Create Project**.



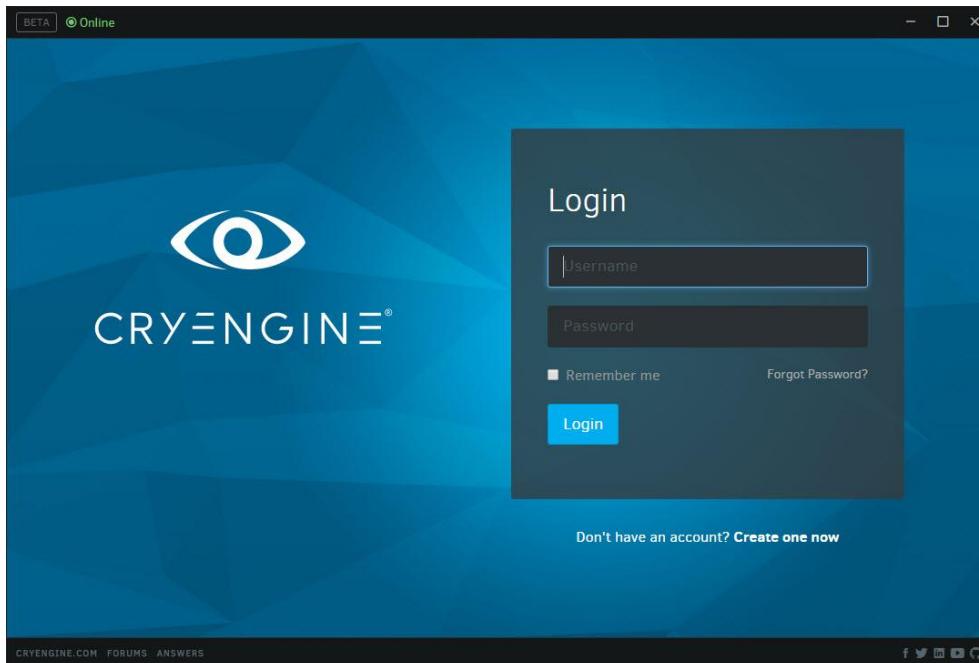
7. Your project will be saved and it will appear in the list of projects as seen in the image below.



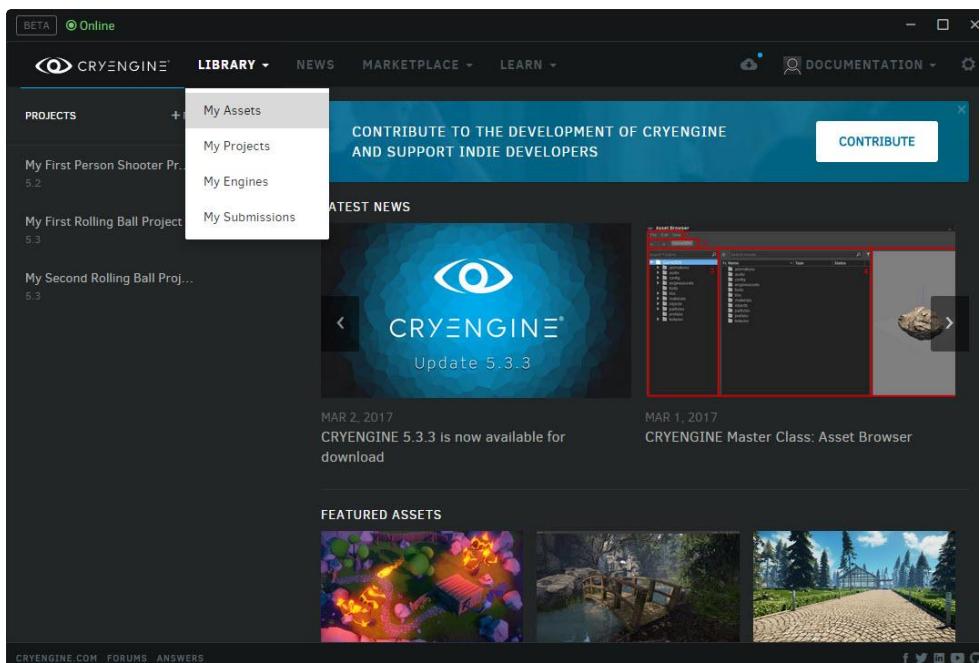
## Adding Assets to Your Project

The CRYENGINE Marketplace offers developers a comprehensive library of AAA quality, ready-made assets that can be used to personalize a project. You can choose from a wide range of high quality, professionally crafted, and curated content, all available for download now (CRYENGINE account required). The Marketplace is available from [www.cryengine.com](http://www.cryengine.com) or you can click [here](#) and log in using your CRYENGINE account.

1. Open the CRYENGINE Launcher from the icon on your desktop and then log in to your account.

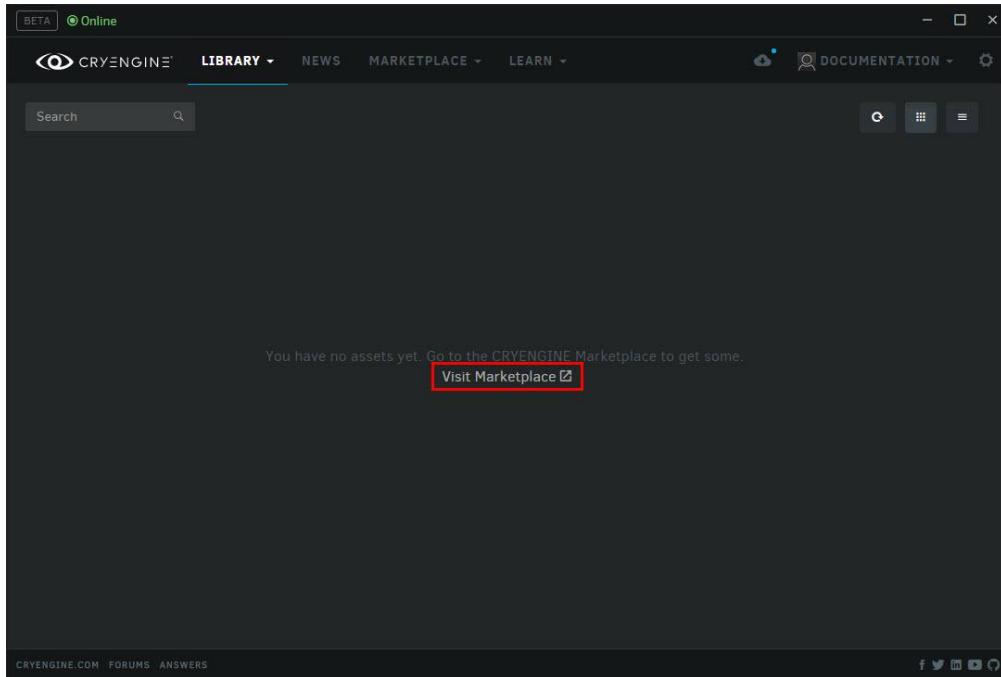


2. In the Launcher main window, click on **My Assets** under the **Library** menu.

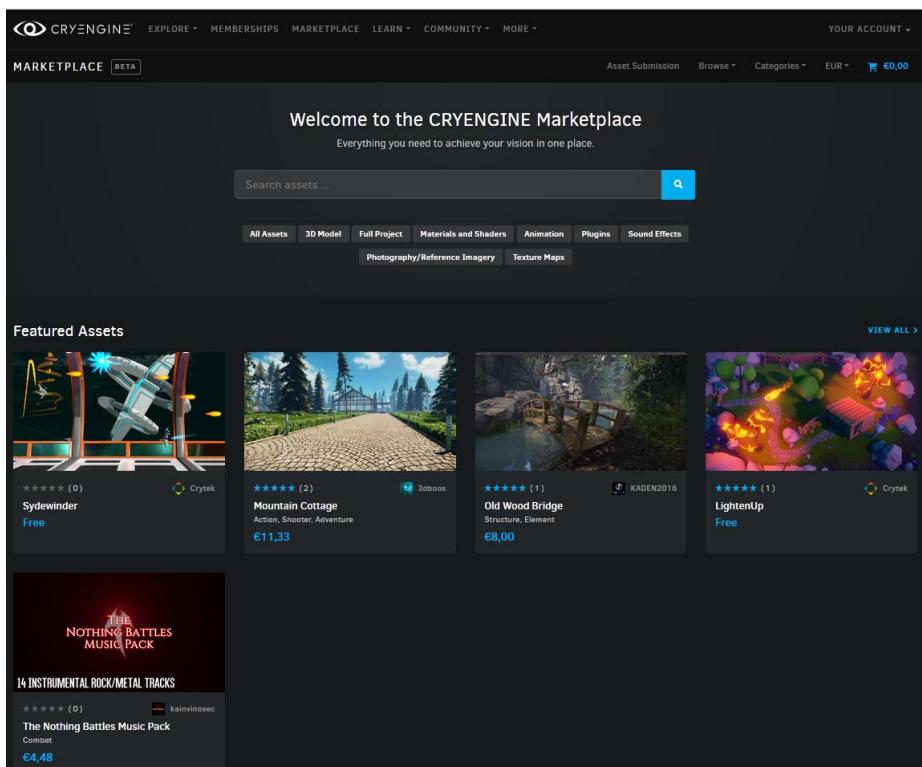


# Getting Started With CRYENGINE Guide

- In the **My Assets** window, click the **Visit Marketplace** option to open the CRYENGINE Marketplace website. You can also get to the Marketplace from [cryengine.com](http://cryengine.com)



- Assets can be found using the **Search assets...** option or through any of the asset category options. In the example below we want to download the CRYENGINE GameSDK Sample Project. This particular asset is located in the **Full Project** section.



# Getting Started With CRYENGINE Guide

- In the example below we have decided to download the CRYENGINE GameSDK Sample Project. This is a great free sample project that you can use to learn how to create your own first-person games.

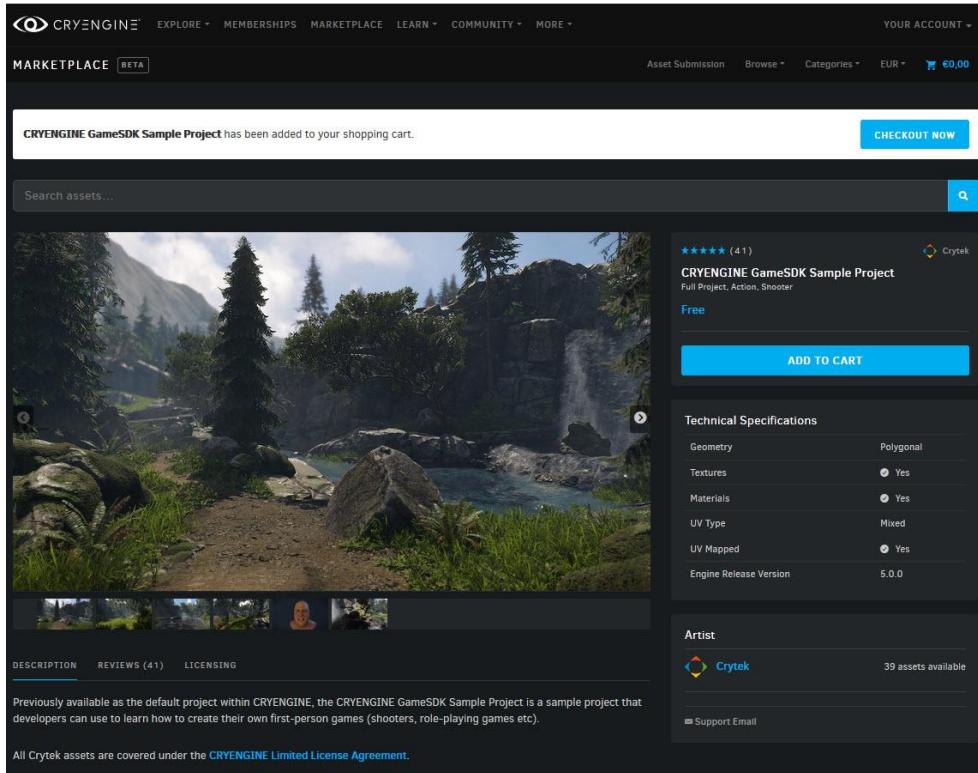
The screenshot shows the CRYENGINE Marketplace interface. At the top, there's a navigation bar with links like EXPLORE, MEMBERSHIPS, MARKETPLACE, LEARN, COMMUNITY, and MORE. On the right, it shows 'YOUR ACCOUNT' with a balance of '0,00'. Below the navigation is a search bar with the placeholder 'Search assets...'. The main area is titled 'All Assets' and shows a grid of asset thumbnails. One thumbnail for 'CRYENGINE GameSDK Sample Project' is highlighted, showing its details: 4.1 stars, Free, Full Project, Action, Shooter, Crytek developer. The sidebar on the left has a section for 'All Assets' with a 'Full Project' filter selected, indicated by a red box. Other filters include Price (with a slider), Rating (with dropdown options for All, 4\*, 5\*, 4.5\*, 4.8\*, and New & unrated), and genres like 3D Model, Materials and Shaders, Animation, Plugins, Sound Effects, Photography/Reference Imagery, Texture Maps, and Training/Tutorials.

- Click on the CRYENGINE GameSDK Sample Project. This opens the Shopping Cart. Then click the **Add To Cart Button**.

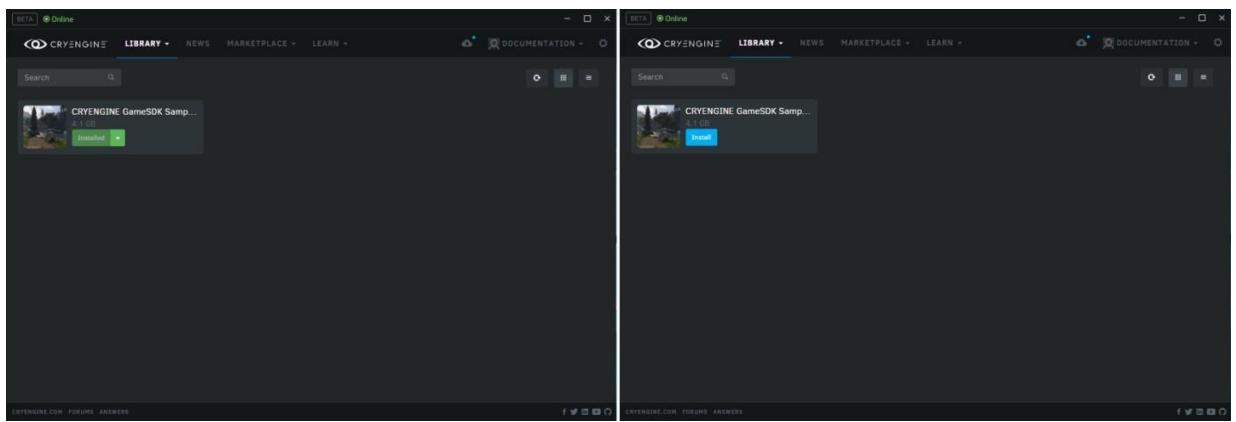
This screenshot shows the product page for the 'CRYENGINE GameSDK Sample Project'. The page features a large image of a forest scene with a path. To the right, the product details are shown: 4.1 stars, Free, Full Project, Action, Shooter, developed by Crytek. Below this is a large blue 'ADD TO CART' button. To the right of the button is a 'Technical Specifications' section with options for Geometry (Polygonal), Textures (Yes), Materials (Yes), UV Type (Mixed), UV Mapped (Yes), and Engine Release Version (5.0.0). Further down is an 'Artist' section showing the Crytek logo and '39 assets available'. At the bottom, there's a note about the project being previously available as a default project and a link to the Limited License Agreement.

# Getting Started With CRYENGINE Guide

7. Click on the **Checkout Now** button.



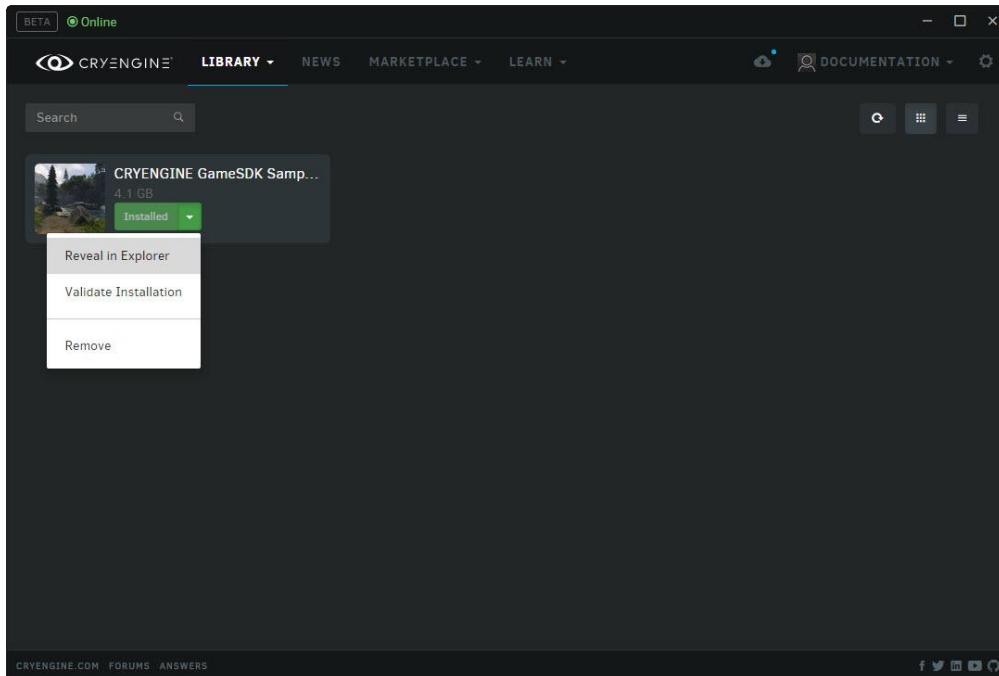
8. You will then need to log in to your CRYENGINE account and follow the on-screen instructions. Once the process is complete, go back to the CRYENGINE Launcher (you may need to refresh it) where the asset will have been downloaded/installed. If the asset has not been installed, click on the **Install** button (second image). Finally, you can confirm installation by clicking on **My Assets** under the **Library** menu - see left image below.



# Getting Started With CRYENGINE Guide

---

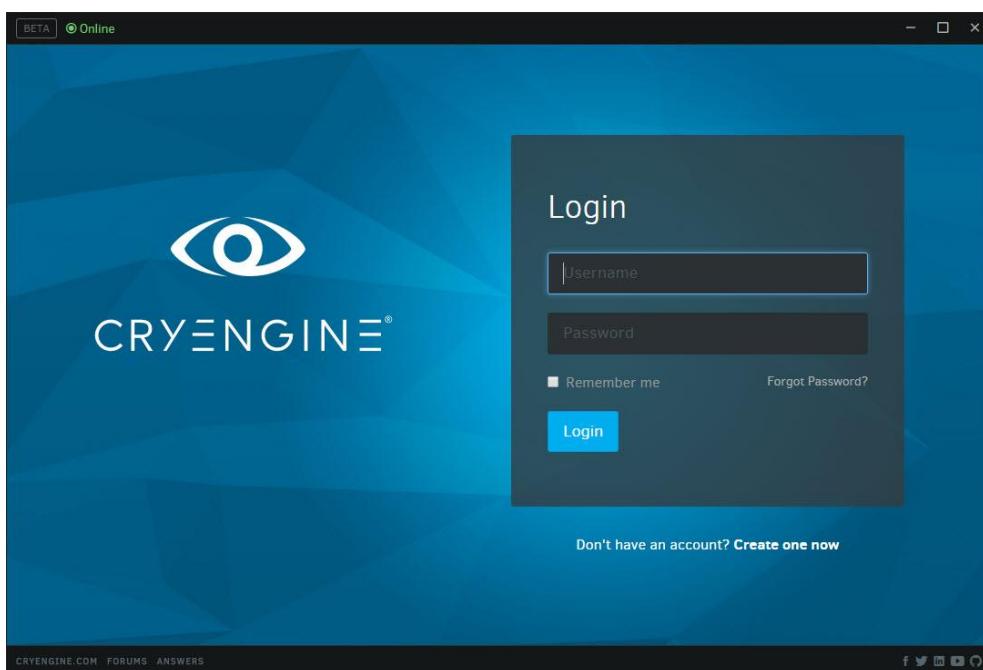
9. **Congratulations**, you have successfully installed your assets into the Launcher. You can use the **Reveal in Explorer** option to view the downloaded assets on your PC and manually move them into your project directory. You can also remove your assets from the Launcher by using the Remove option.



## Importing an Existing Project

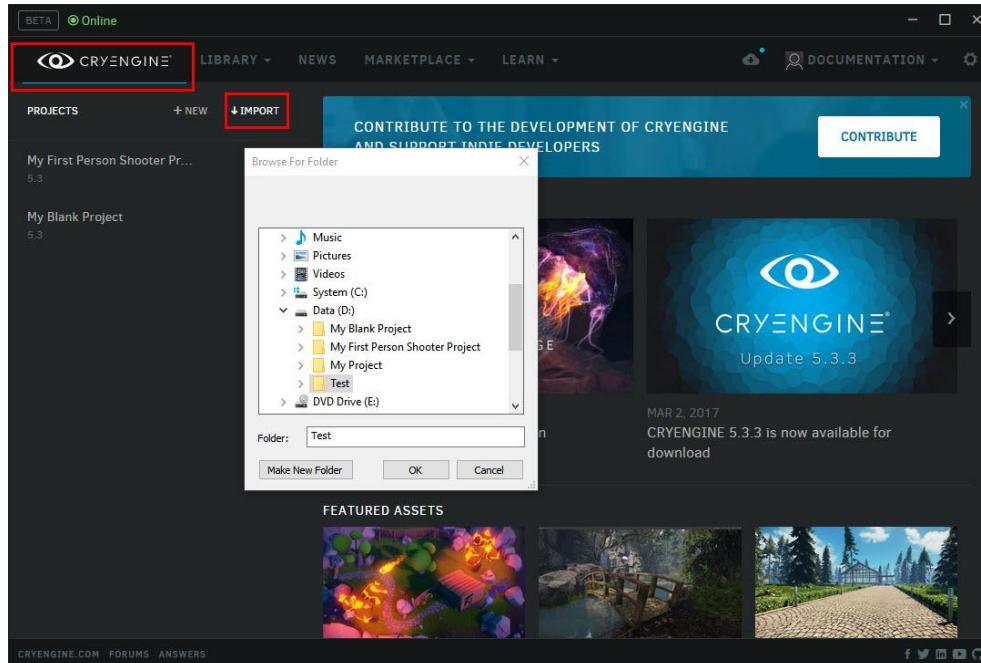
This functionality allows you to import an existing CRYENGINE project into your Launcher.

1. Open the CRYENGINE Launcher from the icon on your desktop and then log in to your account.

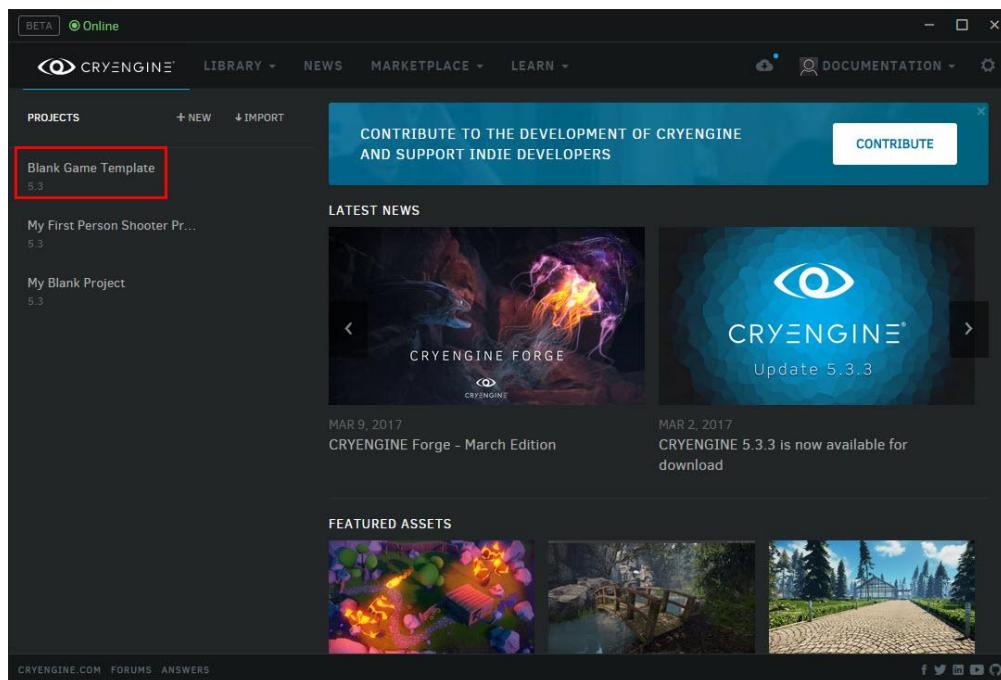


# Getting Started With CRYENGINE Guide

2. In the Launcher main window, under the **Library** menu **My Projects**, click the **+ IMPORT** option. This opens the **Browse For Folder** pop-up where you can navigate to the location of the project that you want to import. In the example below, we will import the project named Test, which is located on the D Drive of our PC. Finally, click the **OK** button.

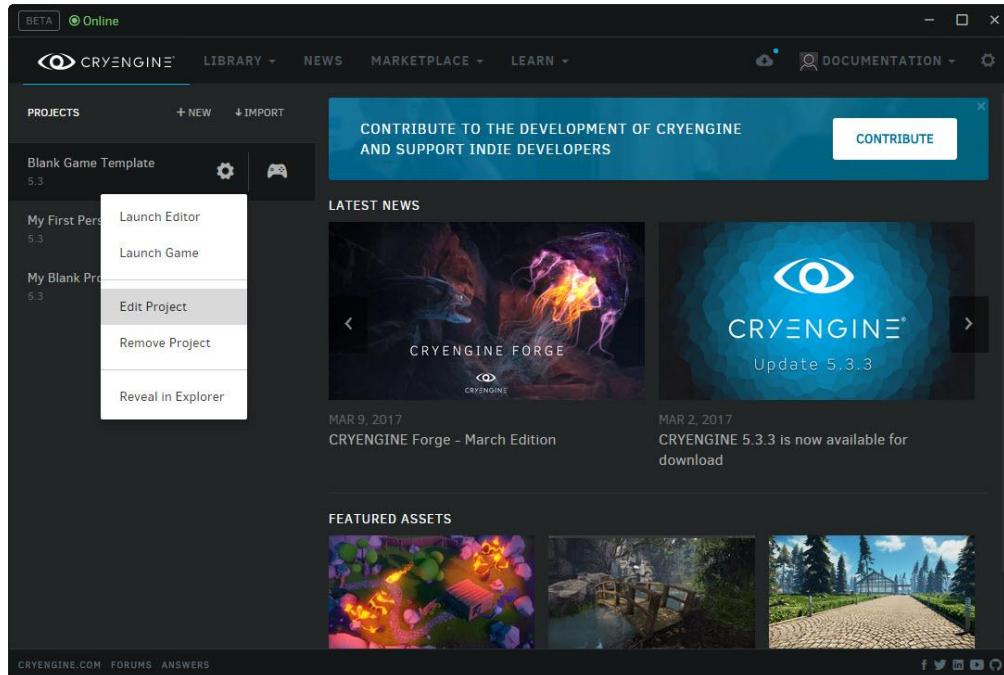


3. The project will then be imported into the Launcher. However in most cases the project's name will have changed. In our example, our project named Test has been renamed to Blank Game Template (this is due to hard-coding within CRYENGINE).

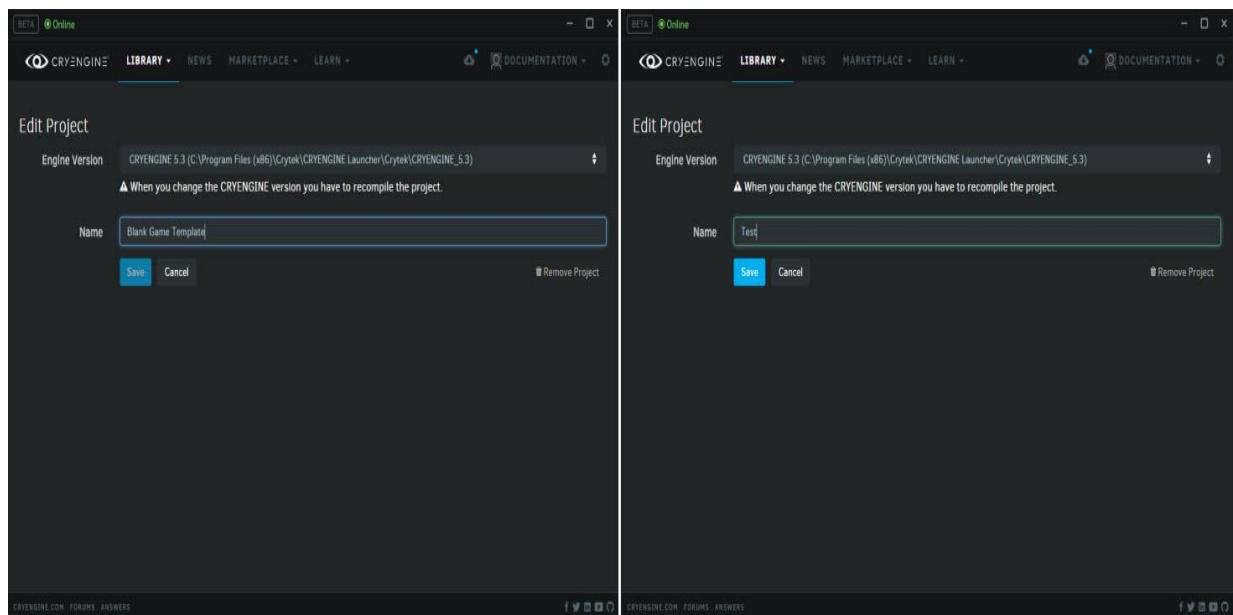


# Getting Started With CRYENGINE Guide

4. Of course, we can change the name back to Test. To do this, mouse over the project in question (in our example Blank Game Template) and click on the cog icon. Then click on the **Edit Project** option.

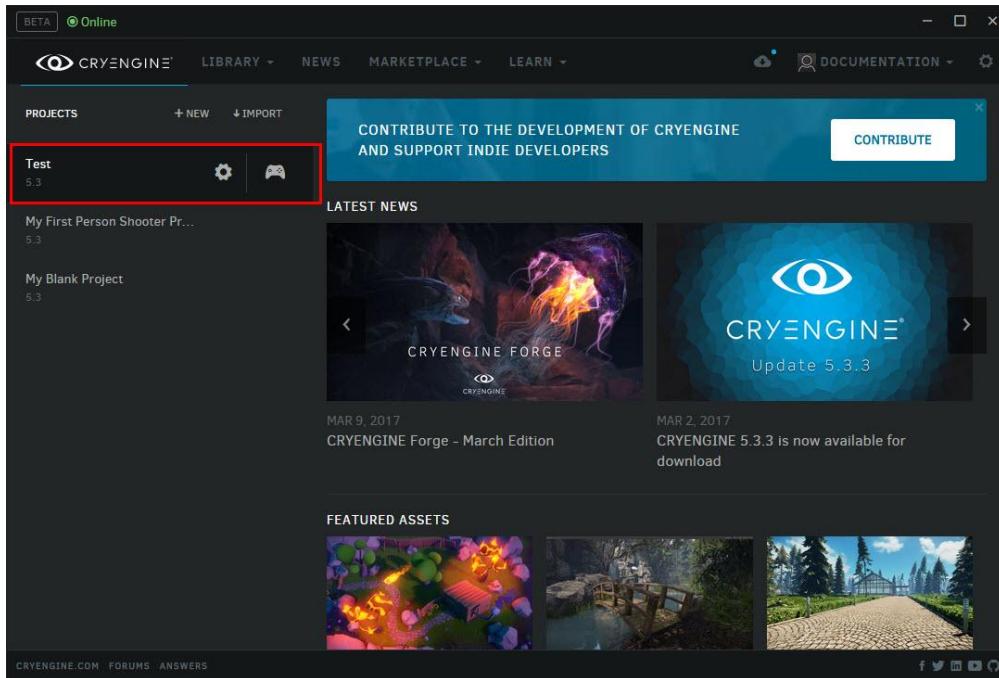


5. This will open the Edit Project window (left-hand image). Change the name of the project (in our example to Test) and click the **Save** button.



# Getting Started With CRYENGINE Guide

6. **Congratulations**, you have now imported an existing project and renamed it accordingly.



## Chapter 5: Installing CRYENGINE Plugins and Tools

CRYENGINE also provides links and tools for various asset creation packages and version control programs. To simplify the setup process, we recommend you use the [CryToolsInstaller](#). You can also setup individual plugins and tools manually.

### CryToolsInstaller

CRYENGINE ships with an installer application (CryToolsInstaller) that lets you install plugins and tools that support the export and content creation processes featured in various Digital Content Creation (DCC) packages. Installation via CryToolsInstaller helps to keep your plugins up to date on each release of CRYENGINE. That means we recommend using CryToolsInstaller rather than manual installs.

**Note:** Before running the installer make sure that you have setup your build information via the **Settings Manager** tool.

### **Running the CryToolsInstaller**

To run the application, navigate to the Tools folder inside your build and double-click on the executable <root>\Tools\CryToolsInstaller.exe

This will open the application. You will then be presented with various options; these depend on which tools are already installed.

# Getting Started With CRYENGINE Guide

---



## Selecting Packages to Install

Next, select the plugin packages that you wish to install. The tools automatically detect the supported application versions and where they are installed on your hard drive.



Once you have chosen your packages, click **Next**.

You will then get a 'success' message (or a 'fail' message if something has gone wrong) on the installation of the plugins.

# Getting Started With CRYENGINE Guide

---



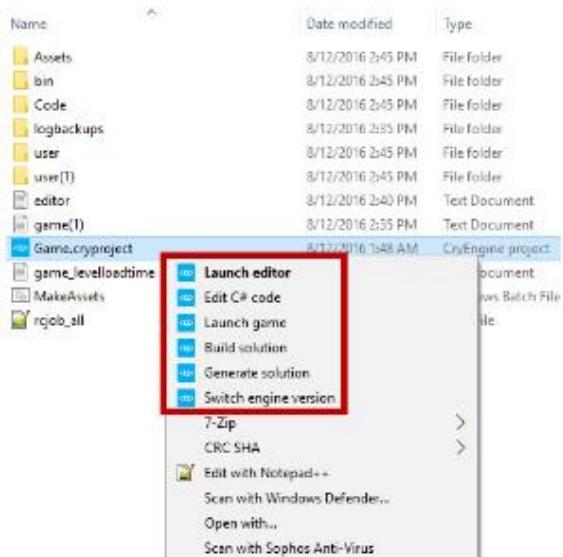
# Getting Started With CRYENGINE Guide

---

## Project Launcher Tools

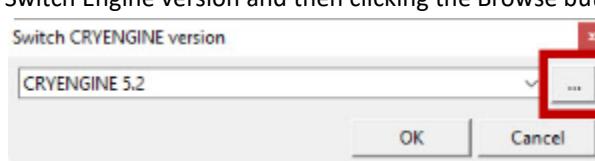
In your project folder you will find CRYENGINE project files - the filename extension for these types of file is \*.cryproject

When you right-click on these files you will get a number of options to easily perform actions.



These options do the following.

Options	Description
Launch Editor	Launches the Sandbox Editor associated with the project. You can open a map from your project by going to <b>File -&gt; Open</b>
Edit C# Code	Opens MonoDevelop so you can edit the C# code of the project.
Launch Game	Launches the game.
Build Solution	Builds the project binaries (game.dll) from the code solutions. <b>Note:</b> This requires Visual Studio to be installed. For C# projects MonoDevelop will also need to be correctly set up.
Generate Solution	Generates your code solution for C++ and C#. The solution will need to be regenerated if the project is moved to a new hard drive location. <b>Note:</b> This requires CMake to be installed; otherwise you get an error message saying "Unable to locate CMake". You can download CMake <a href="#">here</a> . You also need to install Visual Studio and create a C++ project in Visual Studio so that the C++ compiler is installed.
Switch Engine Version	Lets you change the Engine version that you use for the project. The Engine versions in the drop-down menu are the versions that you have installed. If you have downloaded an Engine version from GitHub, you can select it here by choosing Switch Engine version and then clicking the Browse button in the menu that appears



# Getting Started With CRYENGINE Guide

You can then browse to the location where you've saved that Engine version.

**Note:** This simply needs to point to the root folder of that CRYENGINE version. This will only work with Engine version 5.2.0 and later releases.

## Chapter 6: Sandbox Editor Overview

The feature systems found in CRYENGINE allow you to create stunning experiences in your games. In order to use these systems efficiently it's good to have a basic understanding of the overall User Interface (UI) in the Sandbox Editor.

**Note:** The CRYENGINE Sandbox Editor is often referred to as just the Editor, the Sandbox or written in full as the Sandbox Editor.

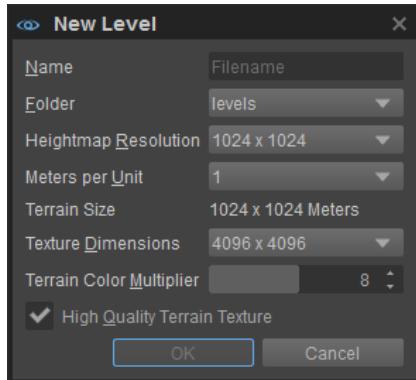
For more information regarding the basic features and navigation in the Editor then please refer to the topics below.

- [Creating a New Level](#)
- [CRYENGINE Objects](#)
- [Editor Layout](#)
- [Essential Tools Editors](#)
- [Level Explorer](#)

### Creating a New Level

A level depicts an area or a map that will be a part of the game where actual game development takes place. A game usually consists of multiple levels that are merged to form a storyline. To begin, let's create a new level using the Sandbox Editor.

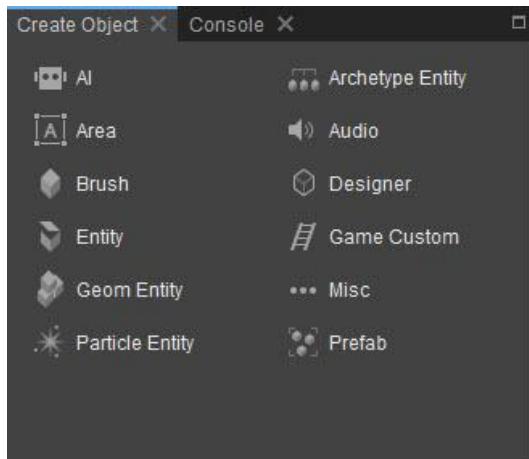
1. Start the Editor and create a new level folder using the **File -> New** option.



2. Enter a name for your new level along with the specified parameters shown in the above window.
3. Based on your requirements, generate a terrain for your level; you can specify your Heightmap Resolution and Texture Dimensions in the New Level window. A heightmap is used to store the game data for your level.
4. Click **OK** to load your finalized level.

## CRYENGINE Objects

The Create Objects tool provides access to create all the Object types within the CRYENGINE Sandbox Editor. The following objects are available.



Options	Description
AI	AI control objects are used to control AI entities and their behaviors in the game world. They define a specific behavior for an AI with reference to its location.
Archetype Entity	Based on the regular Entity, this specifies individual parameter values for that Entity. If the value of an Archetype parameter is changed, all instances of that Archetype in the scene will be updated automatically.
Area	Used to create three dimensional zones in the level that can be used to trigger events.
Audio	Used to place an audio file in your level.
Brush	Brushes are static objects placed in the world that cannot be modified or moved dynamically during gameplay.
Designer	Used to create complex meshes with powerful built-in functionality, without the need for any external DCC tools.
Entity	Enables placing game specific objects in your level.
Game Custom	Game Custom objects are entities used within the Sandbox that are specific to a particular game or project.
Geom Entity	Allows entities to fetch the physical parameters from its assigned geometry.
Misc	Comprises tools and functions used in game development and level design.
Particle Entity	Particle Entity is used by Particle Effects in the level created in the <a href="#">Particle Editor</a> . It acts as a 'container' for a particle effect and it can be attached to any object using the link feature.
Prefab	Prefabs are groups of objects that can be placed in the level as instances. An instance is an object that is an exact copy of every other object of the same type. Altering one prefab universally applies the changes to each instance of the prefab object.

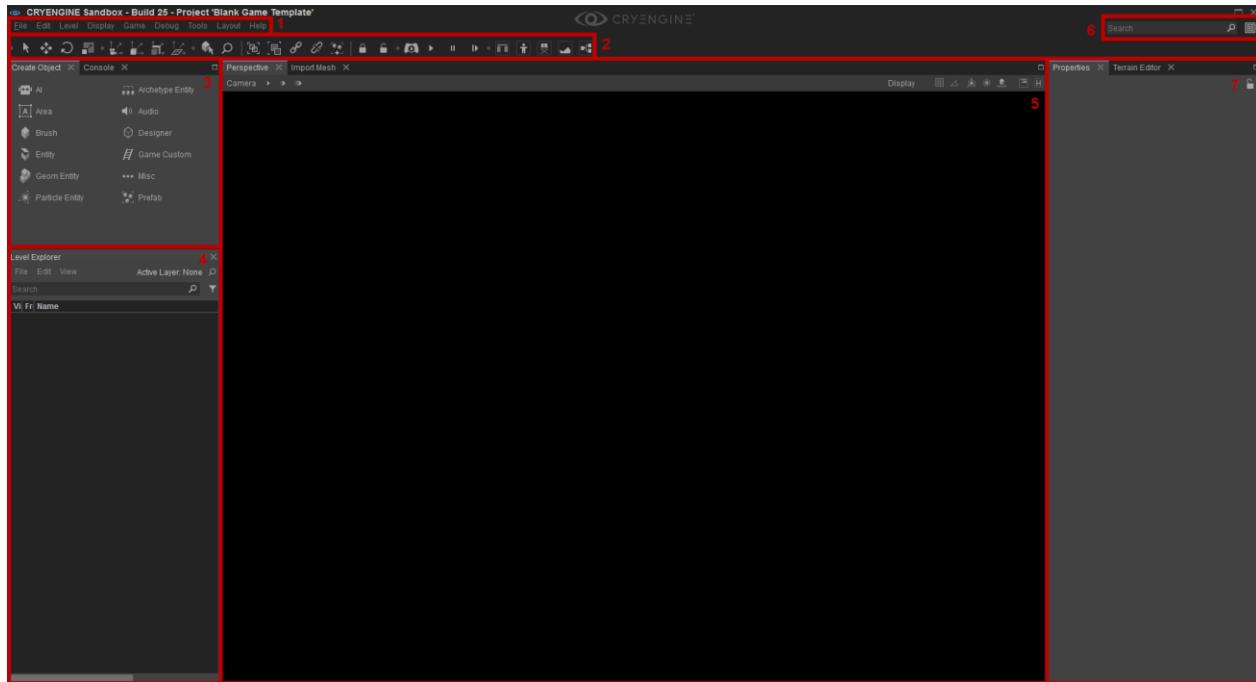
# Getting Started With CRYENGINE Guide

---

The Create Object tool is shown in the Sandbox interface by default, but can also be found in **Tools -> Level Editor -> Create Object**.

## Editor Layout

By default, the CRYENGINE main window comprises of the following parts.



1. Menu Bar
2. Tool Bar
3. Create Object/Console
4. Level Explorer
5. Viewport
6. Menu Search
7. Properties/Terrain Editor

## Essential Tools Editors

To use the basic tools in the Sandbox Editor, you need to first place an object.

### Placing an Object in Your Level

From the **Create Object Console**, click **Brush** to display the assets you can add to the level.

Under the directory tree, navigate to an object.

You can also preview the object while selecting it.

# Getting Started With CRYENGINE Guide

---

**Note:** You will first need to import the objects (download them from the Marketplace) into your project folder so that they can be used in a level. For example, `YOUR_PROJECT_FOLDER\Assets\Objects`.

See [this page](#) to find out how to import assets to your Launcher

1. Drag the object into the Perspective viewport.
2. You have successfully placed an object into your level.

The CRYENGINE Sandbox Editor features the following basic tools to manage objects within a level.

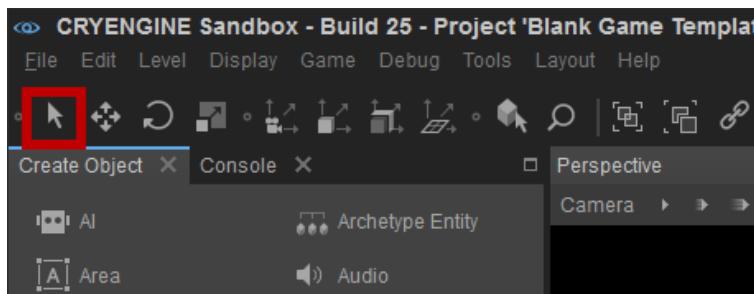
- Select (Shortcut key: 1)
- Move (Shortcut key: 2)
- Rotate (Shortcut key: 3)
- Scale (Shortcut key: 4)
- Coordinates System
- Snap Features
- Navigation key shortcuts
- Switching to Game Mode

## Transforming Objects

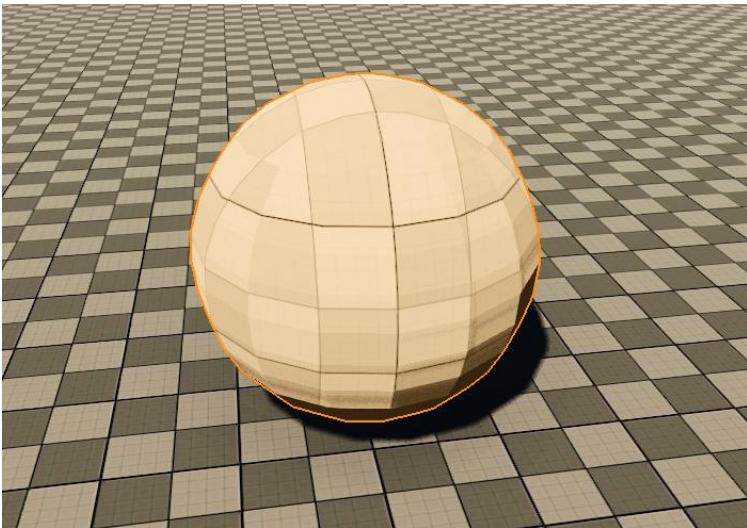
CRYENGINE features a variety of ways to transform objects. On this page, you will learn the basics of how to use the different modes at your disposal.

### **Select**

In Select mode you can select single objects, a group of objects, or add to or subtract from the current selection in the Viewport window with simple mouse actions. Note: Select mode literally only selects objects without moving them.



Using Select mode you can choose any object in the Perspective Viewport. Select mode can be activated by clicking on the button in the toolbar, using the Keyboard Shortcut '4', or via **Edit -> Editing Mode -> Select Mode**.

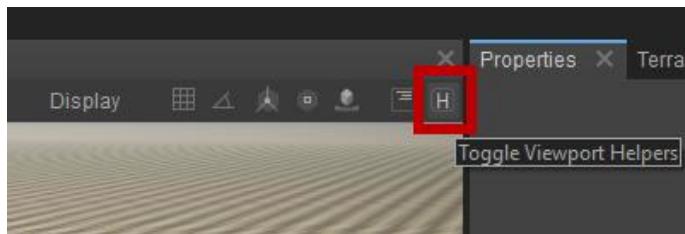


## Selecting a Single Object

Objects with physics meshes can be selected by moving the mouse cursor over the object and clicking the Left Mouse Button (LMB).

You can select 3D objects (without physics mesh proxies built into the asset) by clicking on it. The edges of the object will be visible when you click on it.

**Tip:** Turn Helpers on (**Ctrl+H** or the **Toggle Viewport Helpers** button in the top-right corner of the viewport) to see the edges while you mouse over objects:



2D Elements like particle entities can be selected via a 2D helper icon displayed in the objects origin. These icons are toggled on/off with the **Show Helpers** button (see above).

## Selecting Multiple Objects

To select multiple objects, drag a box around the objects that you want to select, then release the mouse button when all the required objects are inside the selection box. When using drag select, it grabs everything within that 2D box to infinity. In busy scenes with a lot of assets, it helps to angle the camera to reduce the number of assets on screen (i.e. look towards the floor or the sky).

You can also hold the **Ctrl** key and select every object one at a time if you want to be more specific with your selection.

The current version of CRYENGINE lets you transform your selected objects using a single gizmo, even when you have selected multiple objects in the Viewport. This helps to reduce gizmo clutter when multiple objects are selected in the Viewport.

**Note:** Gizmos only appear in Move, Rotate and Scale mode, not in Select mode. The gizmo will appear at the central point of the selected objects and when a local or parent coordinate frame is used, the transform gizmo will use the relevant coordinate frame of the last selected object.

# Getting Started With CRYENGINE Guide

---

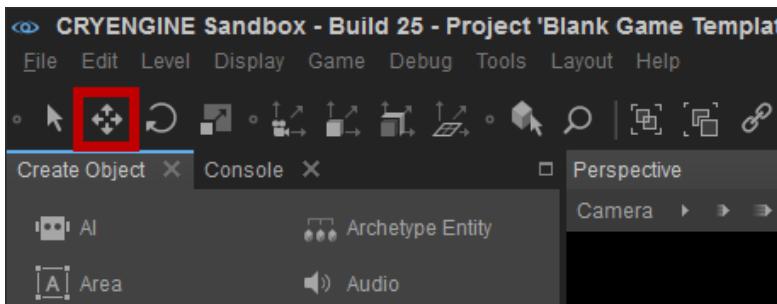


## ***Adding and Removing Objects from Selection***

1. Hold down **Ctrl** and click the object you want to add or remove from the selection.
2. You can add multiple objects at a time by holding **Ctrl** while dragging a box around the objects.

## ***Move***

The **Move** mode lets you select objects and move them around in the level.



This can be done by click-dragging (**LMB**) on the handles of the Axis Gizmo or by first selecting an axis and then just click-dragging (**LMB**) on the object.

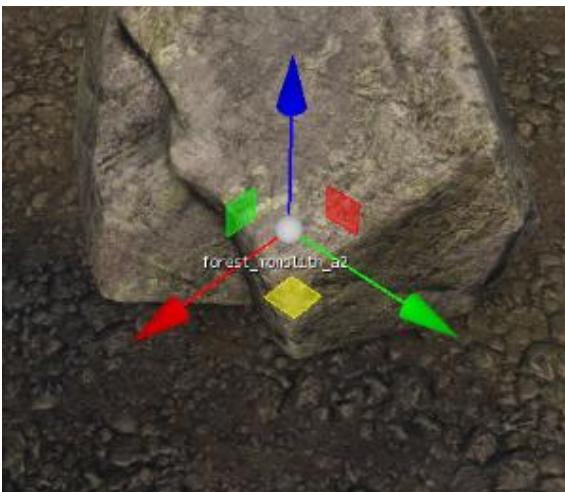
The transformation will then be performed according to the currently activated transformation constraints.

In order to fine tune the interactive translation, you can activate [Grid snapping](#), [Snap to Terrain/Snap to Geometry](#), Axis Constraints (see below) or adjust the [Coordinate System](#) that you want to use for the translation. Move mode can be activated by clicking on the button in the toolbar using the Keyboard Shortcut '**1**', or via **Edit -> Editing Mode -> Move**.

In Move mode the Axis Gizmo has an arrow at the end of each axis.

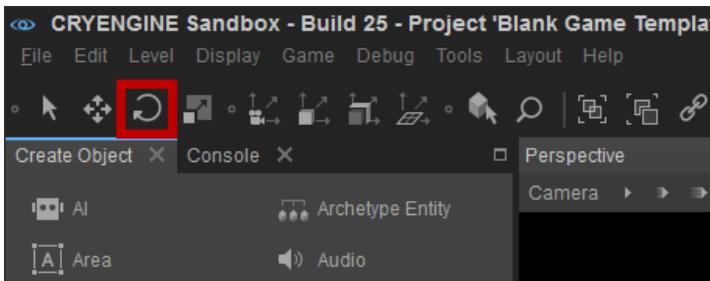
# Getting Started With CRYENGINE Guide

---



## **Rotate**

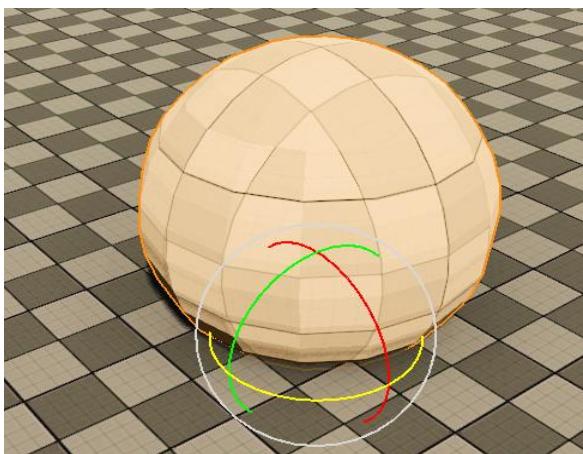
Rotate mode lets you select objects and rotate them interactively in the Viewport window.



This can be done by click-dragging (LMB) on the Axis Gizmo handles or after locking the selection by click-dragging (LMB) on the object. The rotation will then be performed according to the currently activated transformation constraints.

Rotate mode can be activated by clicking on the **Rotate** button using the Keyboard Shortcut '2', or via **Edit -> Editing Mode -> Rotate**.

In Rotate mode the Axis Gizmo consists of several (semi-)circles.



# Getting Started With CRYENGINE Guide

---

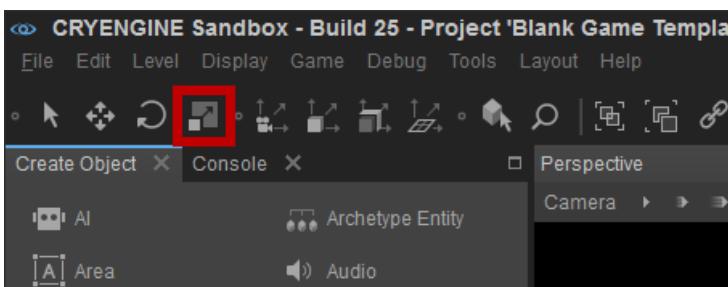
The red, green and blue (yellow in the image above because it's selected) wheel gizmos correspond to the X, Y and Z axes, while the white wheel always rotates the object in screen space.

When you use the wheel gizmos to rotate the object, you can also see the angle of rotation at which the object is being rotated.



## Scale

Scale mode lets you select objects and scale them interactively in the Viewport window.



Objects can be scaled proportionately and in every direction by click-dragging (LMB) on the **center cube** of the Axis Gizmo (this keeps the physics mesh intact).

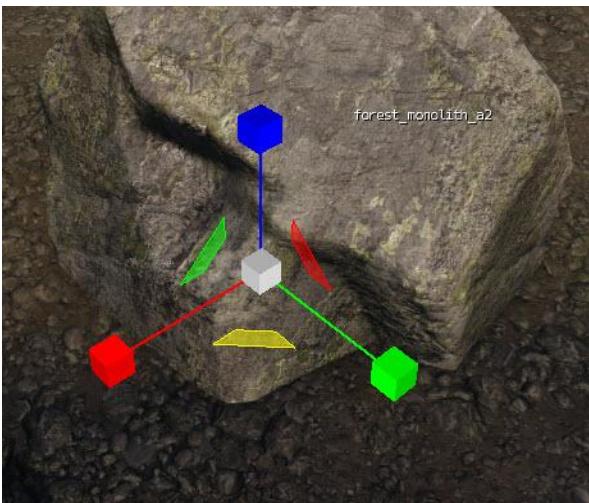
Non-uniform scaling can be achieved by click-dragging on one of the Axis Gizmo handles; this also works when you have selected multiple objects and when you have locked your selection. Finally, in order to fine tune the interactive scaling you can activate Axis Constraints (see below), or adjust the [Coordinate System](#).

Scale mode can be activated by clicking on the button in the toolbar, using the Keyboard Shortcut '3', or via **Edit -> Editing Mode -> Scale**.

In Scale mode the Axis Gizmo has a cube at the end of each axis.

# Getting Started With CRYENGINE Guide

---



Similar to Move mode, you can click and drag the cubes to scale the object in one direction, or you can click and drag one of the square-shaped plane gizmos between the Axis Gizmos to move the scale along two axes at the same time.

**Note:** Be careful when doing this, as non-uniform scaling does not affect the physics mesh. Player collision or hit detection may no longer work as expected.

Finally, you can click and drag the white cube in the middle to scale the object along all three axes at the same time.

**Note:** When you scale an object, the gizmo displays how much the object is scaled (compared to its original size)

## Coordinates System

CRYENGINE by default features a world coordinate system where the Z-Axis points upwards. When manipulating objects directly with the Axis Gizmo you can change the coordinate system to help with manipulation.

You can also change the coordinate system directly from the Coordinates toolbar.



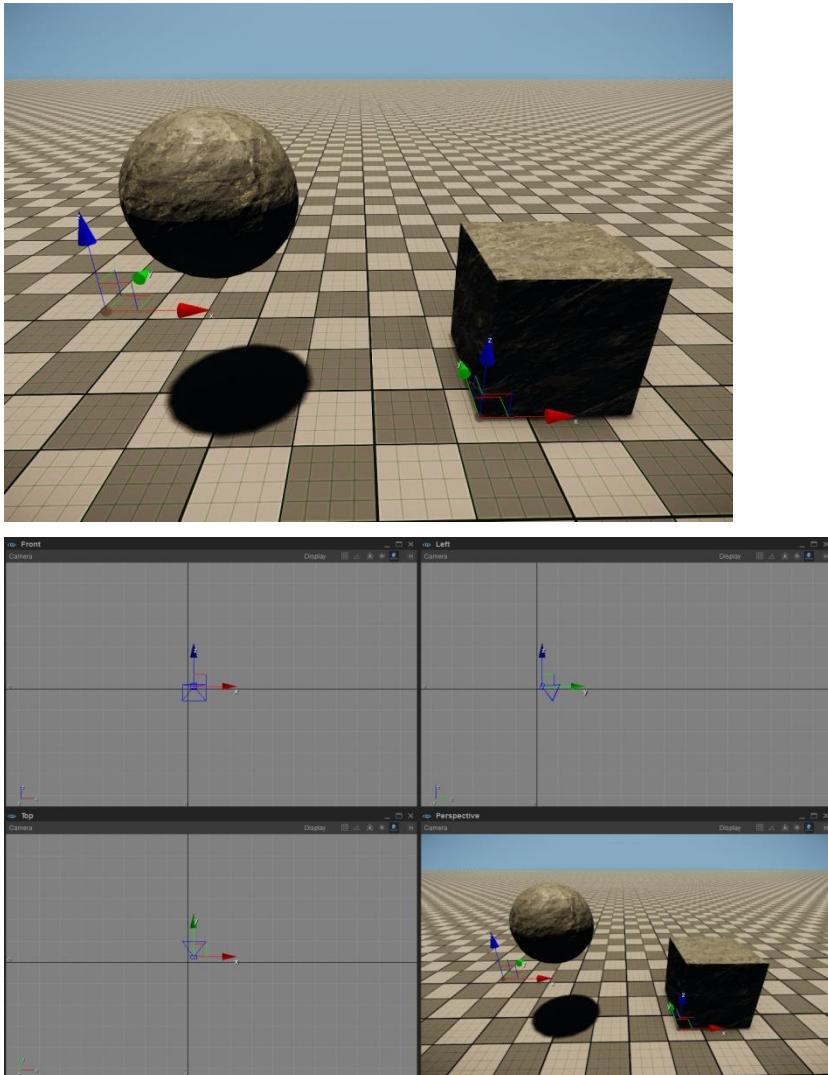
#	Coordinate System	Description
1	<b>View Coordinates</b>	To be used in the orthographic views, always features an x/y coordinate system.
2	<b>Local Coordinates</b>	Uses the object's pivot for manipulation.
3	<b>Parent Coordinates</b>	Uses the object's parent object's pivot orientation as a reference.
4	<b>World Coordinates</b>	Uses the world axis for manipulation.

# Getting Started With CRYENGINE Guide

---

## ***View Coordinates***

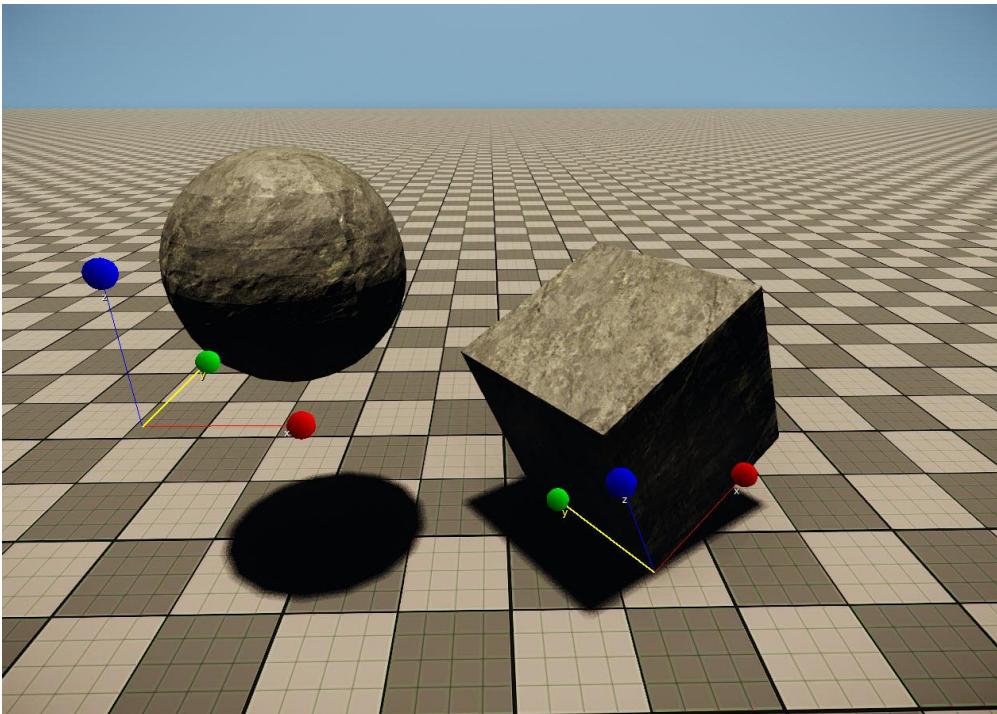
Relevant for Orthographic views. In Perspective view equal to World Coordinate System.



## ***Local Coordinates***

Uses the selected object's original pivot orientation for translation/rotation/scaling.

In the picture below, only the cube has been rotated. Note how the axes are aligned to the cube.



## **Parent Coordinates**

In parent mode, the order of the selection of objects determines the coordinates by which the selection is rotated around. The first object selected becomes the pivot point of which the orientation for translation/rotation/scaling is controlled. This is a moving object concept, so unfortunately it cannot be explained with a still image.

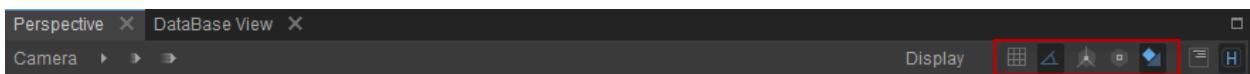
## **World Coordinates**

Uses the world's axis orientation for translation/rotation/scaling. The Axis Gizmo always stays aligned with the world coordinate system. This is a moving object concept, so unfortunately it cannot be explained with a still image.

## **Snap Feature**

When you move an object, you can use Toggle Snapping to Grid to attract that object to points along a customizable grid. **Toggle Snapping to Grid** is set to 'on' by default.

- Toggle Snapping to Grid
- Toggle Snapping to Angle
- Pivot Snapping
- Toggle Snapping to Vertex
- Terrain Snapping



### **To Use Snap Grid**

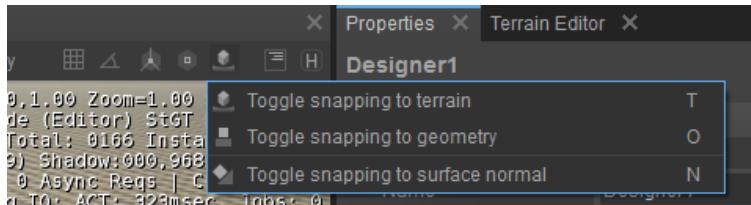
1. To turn grid snap 'off' or 'on', click on the **Toggle Snapping to Grid** button (Perspective window toolbar).

# Getting Started With CRYENGINE Guide

---

2. To customize the size of the snap grid, click and hold the button, then select the preferred value to modify the distance between snap points.

It is also possible to make an object snap to the terrain or all of the geometry. This can be done using the last of the snapping buttons.



With **Snap to Terrain** you can snap an object to the ground. You can toggle this snapping option by pressing **T**.

**Snap to Geometry** snaps the current selection to any piece of geometry in the level, which includes the terrain and any objects and regardless of their position in the level. For example, you could also snap an object to another object hanging in mid-air. You can toggle this snapping option by pressing **O**.

## Basic Navigation - keyboard Shortcuts in CRYENGINE

You can control camera movement with the standard WASD keyboard keys (commonly used in First-Person Shooters) or with the Mouse.

- **W** to move forward
- **S** to move backward
- **A** to move (strafe) left
- **D** to move (strafe) right

Press the **W** or **S** keys to go forwards and backwards and the **A** or **D** keys to strafe left or right with the camera.

Press **SHIFT** together with any **WASD** key to zoom/strafe with **increased speed**.

Roll the **mouse wheel** to move forward or backward.

When the mouse pointer is within the window, hold down the **right mouse button** and move the mouse pointer to rotate the view.

Hold down the **middle mouse button** and move the mouse pointer to pan the view.

## Game Mode

Use the keyboard shortcut (**Ctrl+G**) to switch to Game Mode. This allows you to play (test) the level within the Editor. To exit game mode press **Esc**.

## Chapter 7: Editor Tools

CRYENGINE lets you access various tools and editors for game development. You can access the tools from the **Main Menu -> Tools**.

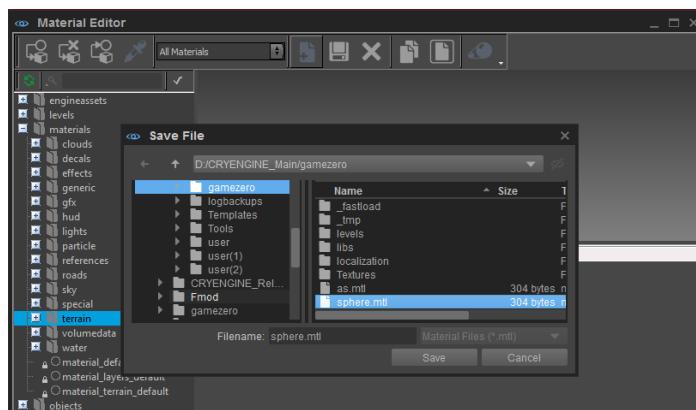
### Materials

With CRYENGINE you can create materials for your level and apply those materials to a designer object. You can also export that object. The following sections provide an understanding of how materials are created and used.

#### *Creating a New Material*

To create a new material, you'll use the **Material Editor**. The **Material Editor** will be mostly used in creating a terrain or an object.

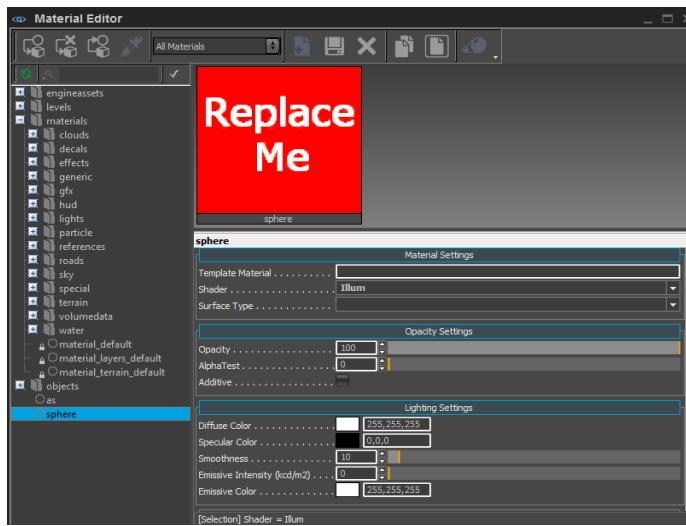
1. To open the **Material Editor** from the main menu, open **Tools -> Material Editor**
2. Under the **Material Editor** window (in the toolbar), click **Add New Item** . You can also add a material by right-clicking the folder (in the Material Editor) where you want to add the material to. A context menu will be displayed where you can select **Add New Material**
3. Navigate to **materials\<any material>**. In the **File name** field type **sphere**. Click **Save**.



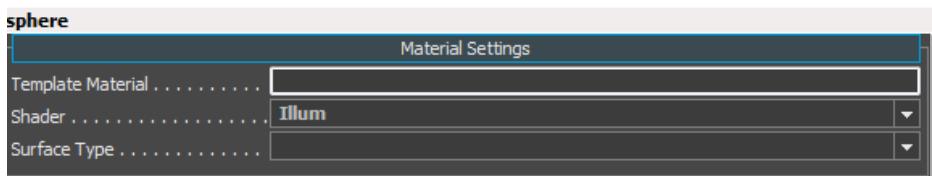
# Getting Started With CRYENGINE Guide

---

4. Select the sphere material you just created. Your **Material Editor** should look similar to the following.



5. In the **Materials Properties and Settings** (lower right area) scroll down and find **Texture Maps**. On the **Diffuse** line click the ellipsis '...' to load a texture file.
6. Under **Material Settings** select any option from the list to be assigned as the **Surface Type**. For example, if you set it as rubber, then it gives the material object the properties of rubber; objects mapped with rubber bounce when hitting another surface.

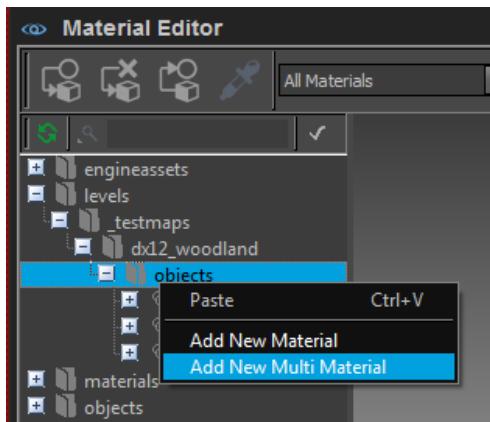


7. Under **Lighting Settings**, set **Diffuse Color**, **Specular Color** and **Smoothness**.
8. Save your material settings. To do this (in the toolbar) click **Save item**.
9. Experiment with the settings to see how adjusting the **Diffuse color**, **Specular color**, and **Smoothness** quickly change the material's appearance.

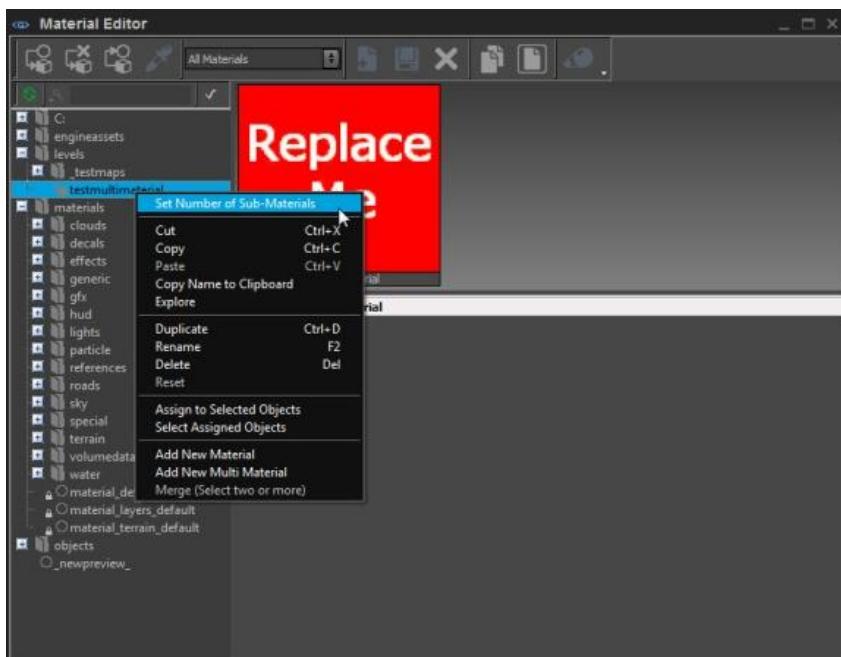
## ***Creating a New Multi-Material***

If you want to have a material with more than one material ID, then you have to create a multi-material.

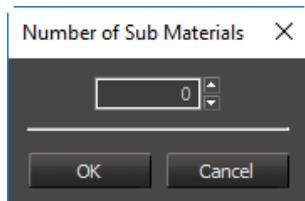
1. Under the **Material Editor** window right-click the folder in which you want to create a multi material and select **Add New Multi-Material**.



2. You now have to set the number of **Sub Materials/Material IDs** that you want to have by right-clicking the material that you have created and selecting **Set Number of Sub Materials**.



3. Specify the number of Sub Materials that you require in your Multi Material file.



# Getting Started With CRYENGINE Guide

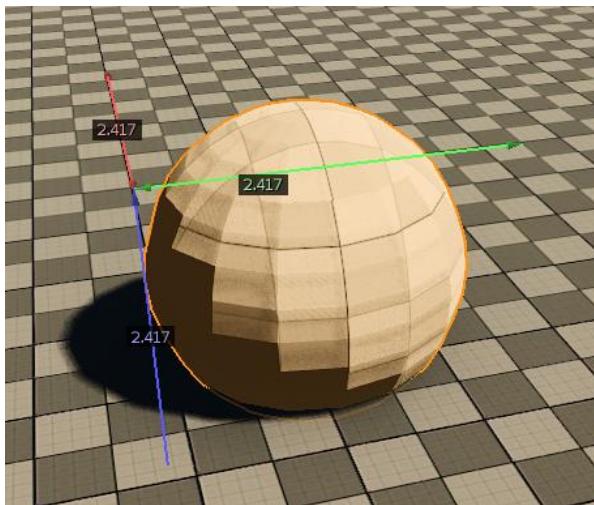
---

4. Right-click each sub material and rename them in a way that is similar to the parent material.
5. Hold down **Ctrl** and select block\_01, block\_02, and block\_03. On the lower right under **Texture Maps** find **Diffuse** and click the ellipsis (...).
6. Assign a texture to the sub materials, and under **Lighting Settings**, set **Diffuse Color**, **Specular Color** and **Smoothness**.
7. Save your material setting by clicking the **Save** icon (in the toolbar).

## **Assigning a Material to an Object**

Now that you have set up your materials you can assign them to objects.

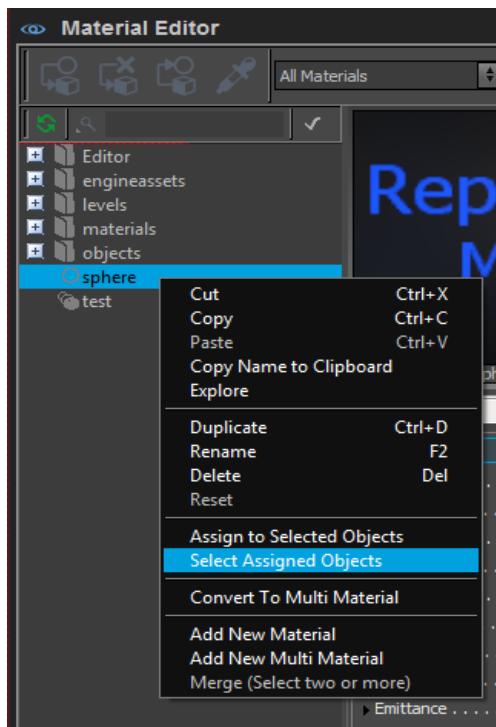
1. First, assign the **sphere** material created earlier to an object created using the **Designer** tool. For example create a sphere. To do this, in the **Perspective** viewport select the sphere object that you created.
2. In the **Material Editor** select the **sphere** material that you created.



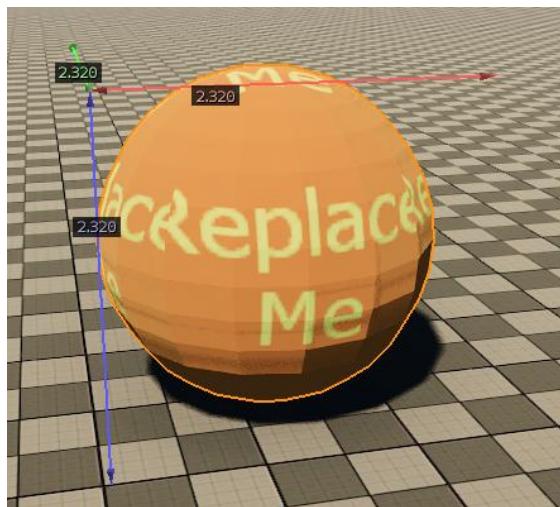
# Getting Started With CRYENGINE Guide

---

3. In the **Material Editor** toolbar, click **Assign Item to Selected Objects** . You can also right-click the material name and select **Assign to Selected Objects**:

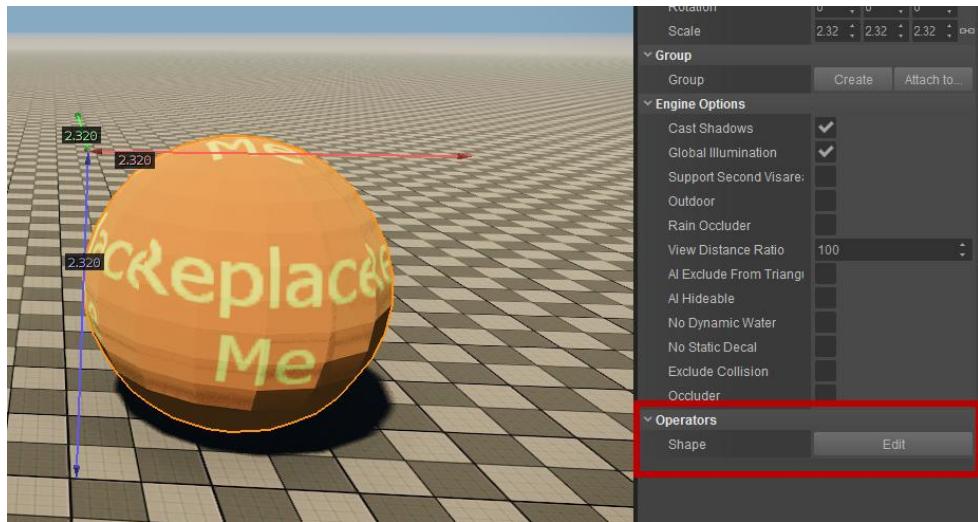


4. After you have applied the material your sphere should look similar to the one in the image below.

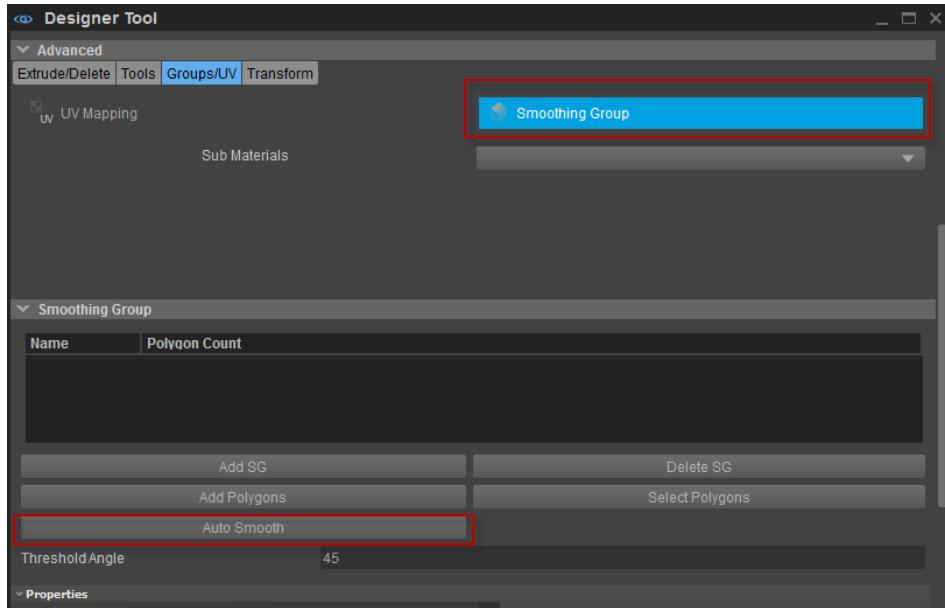


5. Use the Smoothing Group to remove rough edges from the object. To do this, select the object and edit it using the **Designer Tool** by selecting **Tools -> Designer Tool -> Modeling** (see below). You can also navigate to this window by selecting the **Edit** button under the **Operators** section in the **Properties** tab.

# Getting Started With CRYENGINE Guide



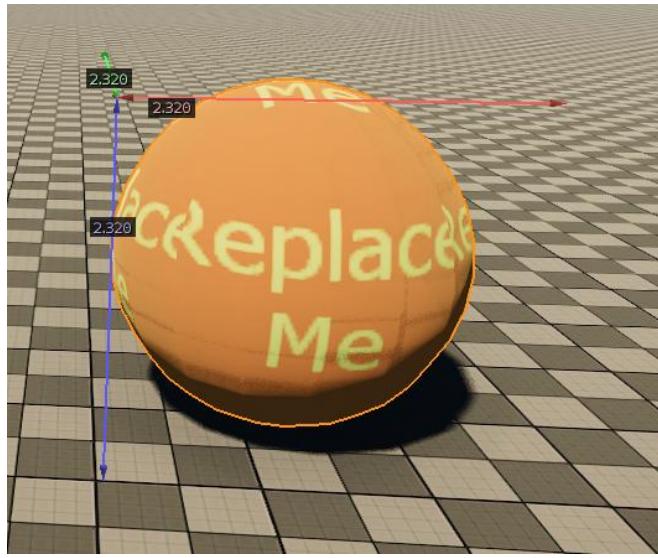
6. In the **Designer Tool** window, under the **Advanced** section, select the **Smoothing Group** button under the **Group/UV** tab.



7. Select all the polygons of the object by dragging a selection box around it.
8. Now, select the **Auto Smooth** button to smooth out the rough edges around the object.

## Getting Started With CRYENGINE Guide

---



9. Congratulations, you have successfully assigned materials to an object!

# Getting Started With CRYENGINE Guide

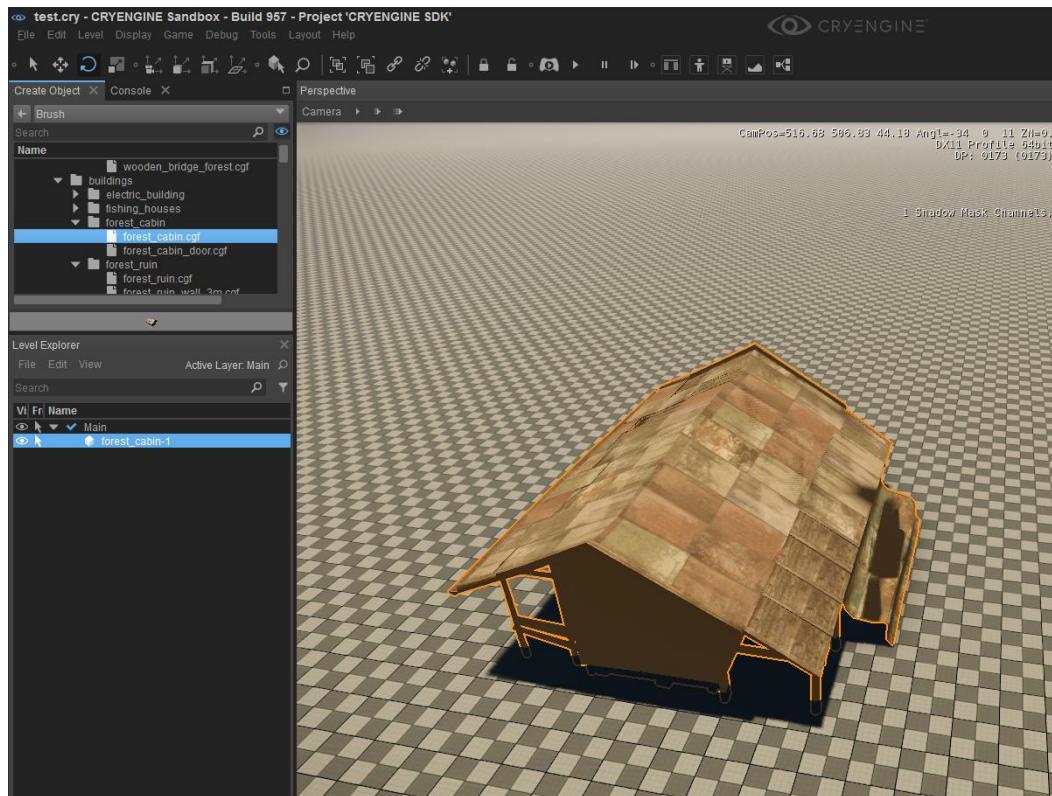
---

## Brushes

In CRYENGINE, static objects are called ‘brushes’. These are ‘things’ that cannot interact with the environment.

**Note:** You need to have downloaded the GameSDK Sample Project from the Marketplace and assigned the assets to your project to find the assets used in this example.

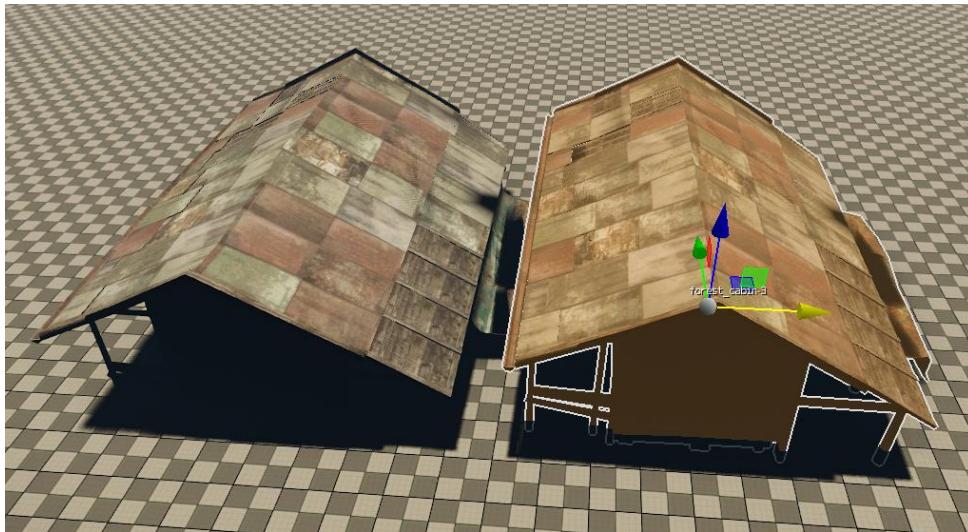
1. In the **Create Objects** tab, click **Brush** to display the assets that can be added to the level.
2. Under **Browser** navigate to the desired brush in the directory tree, in this case under \Game\_Directory\Objects\architecture\buildings\forest\_cabin, we select forest\_cabin.cfg.
3. Drag the object into the **Perspective Viewport** to add it to your level. Click to place your object.



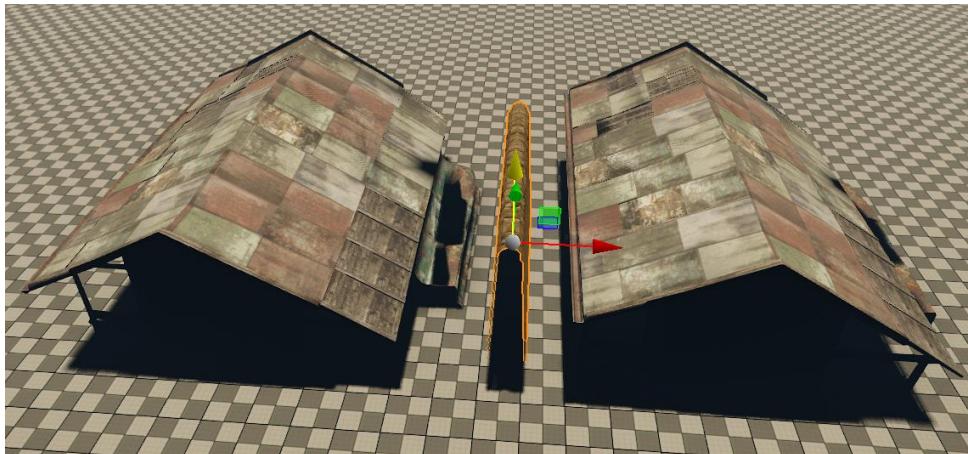
# Getting Started With CRYENGINE Guide

---

4. In the **Perspective Viewport**, select the brush. When selected a gizmo appears at the corner of your brush.
5. Press **Ctrl+C** to copy your brush and **Ctrl+V** to paste it. The copy will be right on top of the original, so move your second brush next to your first brush by using the **Move** tool.



6. Click **Brush** again in the **Create Object** tab.
7. Navigate to \Game\_Directory\Objects\architecture\Walls\farm\_wall\_6m\_high. Drag the wall and place it between the two buildings. Using the basic setup tools such as Move, Rotate and Scale options you can adjust the brushes according to the scene.

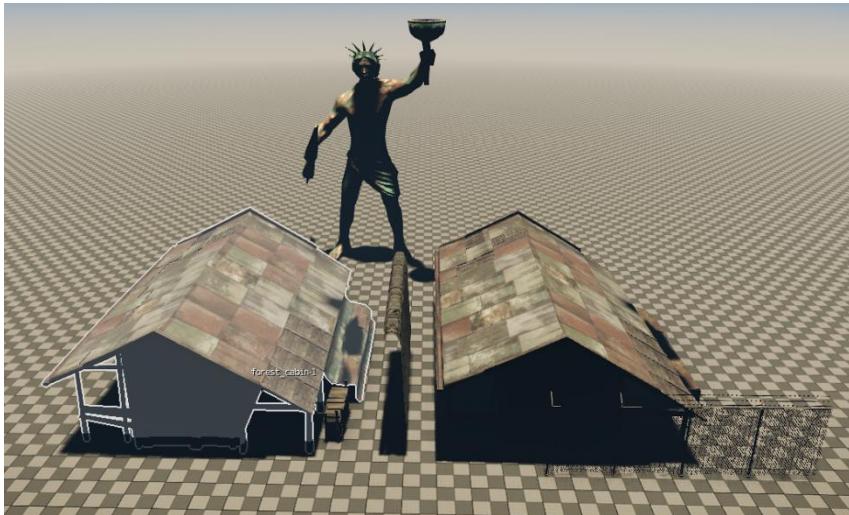


8. Click **Brush** again.
9. Under **Brush**, you can find many other static assets to add to your scene, like a giant statue, or a barbed wire fence. Choose what you want, and don't forget to use the Move, Rotate and Scale tools to put them exactly where you want.

# Getting Started With CRYENGINE Guide

---

Your scene could now look something like the following.



## Terrain

You can use CRYENGINE to apply and modify the terrain of your level. With this tool you can make mountains, valleys, and everything in-between. You can also assign textures and materials to the terrain to make it look like any kind of surface you want.

The Terrain Editor is displayed on the right hand side of the UI by default, but it can also be opened by going to **Tools -> Terrain Editor**.

Using the Terrain Editor you can perform the following functions:

- Generate a new terrain
- Sculpt
- Paint
- Mini Map

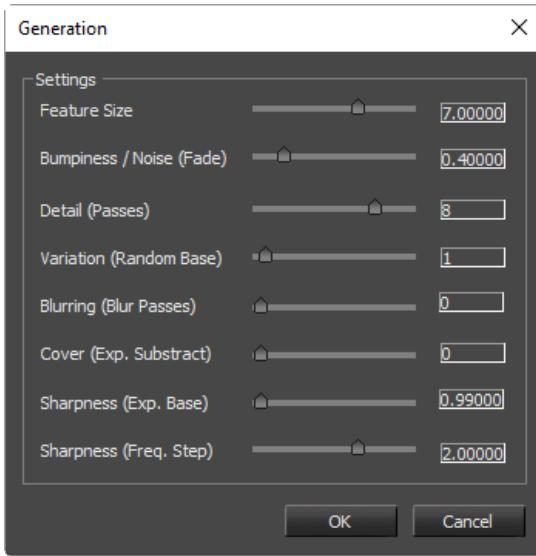
### ***Generating a New Terrain***

You can automatically generate a terrain, which is a very quick way of making a level; however it may require a lot of sculpting work depending on what you want the level to look like.

1. Open a new level from the Sandbox Editor.
2. Under **Terrain Editor** next to the Properties tab in the main window, select **File -> Generate Terrain**.

# Getting Started With CRYENGINE Guide

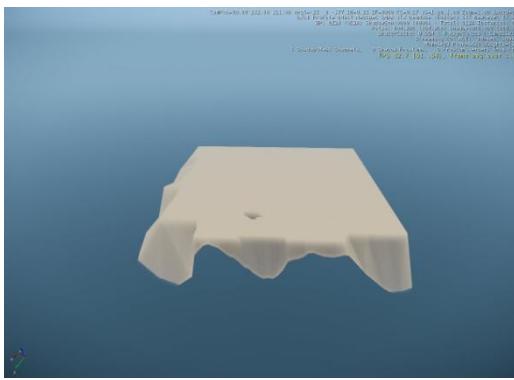
---



3. Specify the settings required for your terrain, and then click **OK** to generate a randomized map. The randomly generated terrain map looks similar to the image below.



4. You can transform this map into a more usable heightmap by selecting **Terrain Editor -> Edit -> Set Terrain Max Height**, and setting the height to a more useable value (height) of say 200.



5. After you have set your heightmap to a more reasonable value, you can generate the terrain again by selecting **Terrain Editor -> File -> Generate Terrain**.

## Sculpting a New Terrain

You now have a new terrain which needs to be sculpted to create a more visually pleasing game environment with features such as mountains, riverbeds etc. Using the **Terrain Editor's Sculpt** feature you can modify the terrain height in various ways by; flattening it, raising/lowering it, making holes/filing holes in it, smoothing it, and moving it.

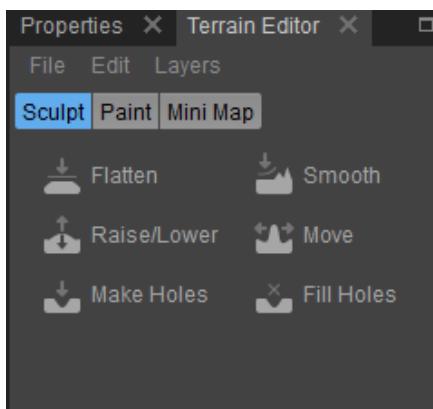
For more information about the Terrain Editor's features please see [this page](#).

You can use the following options to modify the terrain.

- Flatten
- Smooth
- Raise/Lower
- Move
- Make Holes
- Fill Holes

### To sculpt a New Terrain

1. Open the Sandbox Editor and under the **Terrain Editor**, click the **Sculpt** tab.



2. Now activate the **Raise/Lower** button, and select the size of the **Brush** by changing the value in the **Outside Radius** field. You can now see a green circle which is based on the value specified in the above field.
3. You can use the **Ctrl** button to raise or lower the terrain while using the Brush cursor.

Similarly, you can experiment with the Height and Hardness values to find a setup that works for you. Depending on the size of the level and other factors, you'll probably want to use different Hardness and Height values for different situations, so it's important to get a good feel for how this process works.

## Painting a New Terrain

Once you have completed sculpting the terrain, you can start to make it look more realistic by adding vegetation, sand etc. You can use different layers of textures for a nice, natural look with plenty of variation.

**Note:** Before you paint the terrain you need to create textures for it.

To begin with, you need to copy the materials and textures (the ones you downloaded using the CRYENGINE Launcher) to your project folder. You'll find your downloaded assets by opening the Launcher, and selecting

# Getting Started With CRYENGINE Guide

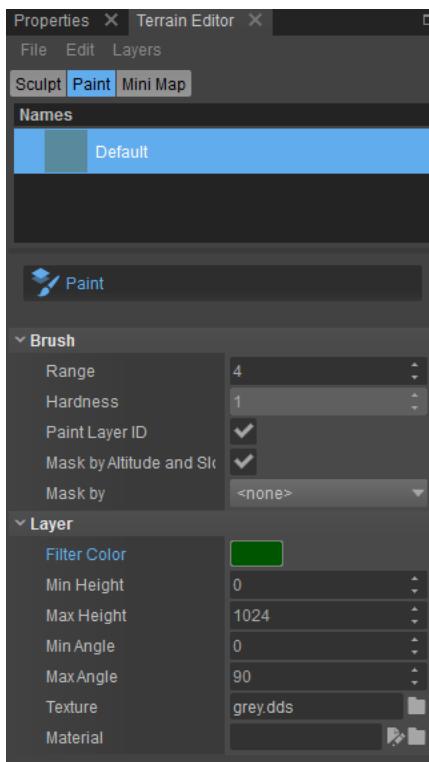
---

**Library -> My Assets.** Under My Assets, select **Reveal in Explorer**, then copy all the \*.pak files and save them to your game folder.

## To Paint a New Texture to the Terrain

In the procedure below we have used a grass texture as an example.

1. In the **Terrain Editor** tab, click the **Paint** option.
2. Under the **Names** section, right-click and then select **Create Layer**.
3. To assign a Material to the texture, click on the **Browse** button next to **Material** in the bottom section of the Terrain Editor.



4. In the **Select File** window, select the material for this layer. You can preview the material using the **Show Preview** button in the top right hand corner of the window, then click **Open**.
5. Now assign a color to the texture by selecting **Filter Color** under the **Layer** section.



6. Set the Layer option according to your level. For example we want the grass to start where the mud ends. In this case we will set the **Min Height** to **19** which will be the Max Height of the Mud layer. We will set

# Getting Started With CRYENGINE Guide

---

the **Max Height** to **26** to ensure that the grass doesn't grow above a certain height. We also need to ensure that the grass isn't painted on the steeper slopes so we have decided to set the **Max Angle** to **50** degrees. **Note:** 0 degrees means a flat surface and we do not want the grass to grow there, so we can leave the Min Angle as it is.

7. After you have provided the values for your layer setting, click the **Paint** button and start to paint the terrain in the Viewport.

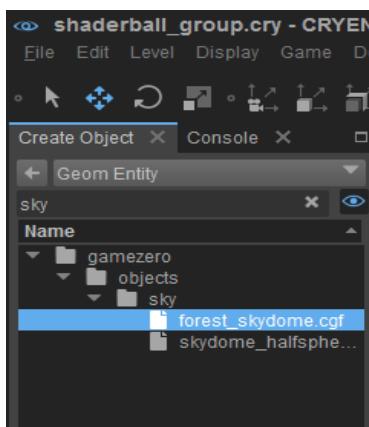
You can also flood the entire terrain by selecting **Layers -> Flood Layer**. Alternatively, you can right-click on the layer and choose **Flood Layer**. The selected layer will then be completely covered in the material and color that you've chosen. However, this is based on the parameters you have set for the height and angles, so only certain areas will be covered in grass.

## Adding Sky to Your Level

To make the game more realistic, you'll want to add sky to your level.

1. Open a level in the Sandbox Editor, under **Create Object** tool select **Geom Entity**.
2. Navigate to **<Project Folder> -> Objects -> Sky**, and choose **forest\_skydome.cfg**.

**Note:** Again, this asset is only available if you have downloaded the GameSDK Sample Project and assigned it to your project. For example, `YOUR_PROJECT_FOLDER\Assets\Objects`. See [this page](#) to find out how to import assets to your Launcher.



3. Drag this entity into your level.

## Lighting

You can light a scene through different lighting features. These are.

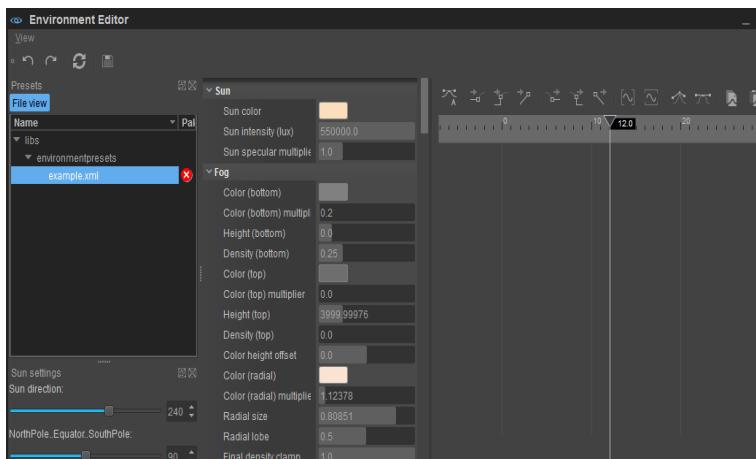
- Time of day
- Environment Probes
- Lights

### **Time of Day**

This feature enables the level to simulate different times of the day which makes shadows longer or shorter etc. Using the Time of Day graph you can show the progression of the values for the selected setting over the course of one day and how they gradually change over time.

#### To Set the Time of Day

1. Open a level in the Sandbox Editor, and then select **Tools -> Environment Editor**.
2. Move the **Time of Day** graph line across the time bar to specify when the sunlight needs to be at its maximum value. You can verify that the time/light changes directly in the viewport window.



3. Specify the **Sun color** and **Sun intensity** values under the **Sun** tab.
4. You can specify various aspects such as **Volumetric clouds**, **Fog** and **Night sky** etc. These can be used to determine the nature of the sky you desire.

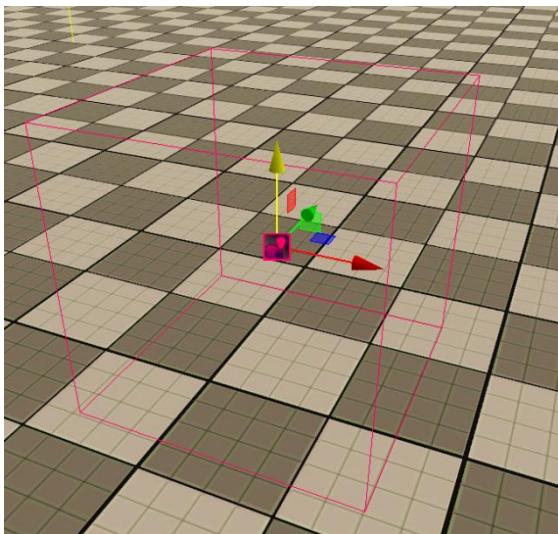
For information on these options refer to the [Environment Editor](#).

## **Environment Probes**

This feature provides indirect lighting to those shadows which are ordinarily too dark. This issue is especially noticeable inside a building, where in a real environment there will always be some indirect sunlight, hence shadows will usually not be completely black. To create some indirect lighting in the most shadowy parts of a level use an Environment Probe.

#### To Add Environment Probes

1. Open a level in the Editor, and then navigate to **Create Object -> Misc -> Environment Probe**. Drag the Environment Probe into your level. Make sure Helpers are displayed – if Move, Rotate or Scale mode is selected, you will notice an Axis Gizmo appears.



2. Select the probe and open the **Properties** tool.
3. Under the **Lua Properties**, set the values for the following parameters **BoxSizeX**, **BoxSizeY**, and **BoxSizeZ**, the values should be based on the area you intend to cover and the size of the level.
4. Now while selecting the probe specify the **Position** values under the **Transform** section.
5. To enable the probe, select **Generate Cubemap** under the **Cubemap** section of the **Properties** tool. If you have more than one probe placed in your level select **Generate All Cubemaps**.

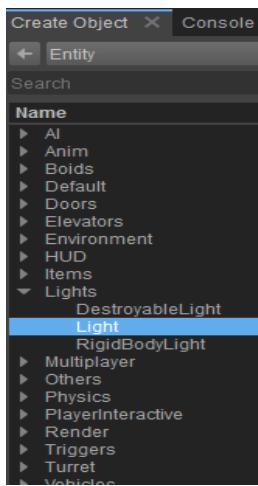
The intensity of a shadow can be changed by selecting the **Environment Probe**. Go to the **Properties** tool then scroll down to the **Entity Properties** section and reduce the value for the **DiffuseMultiplier** (under **Color**).

## **Lights**

By using the Lights entity you can illuminate various features in your level such as adding street lamps or adding lights within buildings. You can also cast shadows for objects within a building.

### **To Add Lights**

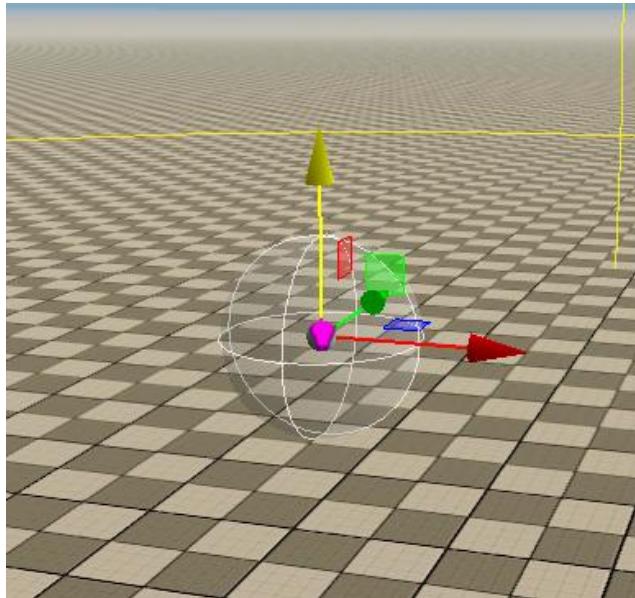
1. In the **Create Objects** tab, click **Entity**, and then navigate to the **Lights** option.



# Getting Started With CRYENGINE Guide

---

2. Select **Light** and drag it into the **Perspective Viewport**.



3. To enable shadows go to the **Shadows** section under the **Properties** tool and change the value from **Never** to **Low Spec** to ensure that shadows are always drawn. You can increase the **DiffuseMultiplier** under the **Color** section to emit more light.
4. To change the light from fill to spot: Under the **Properties** tool, find the **Projector** section. Under Projector select Texture, and then click the folder icon to load a texture from the game folder.

Difference between a house with and without a light object.

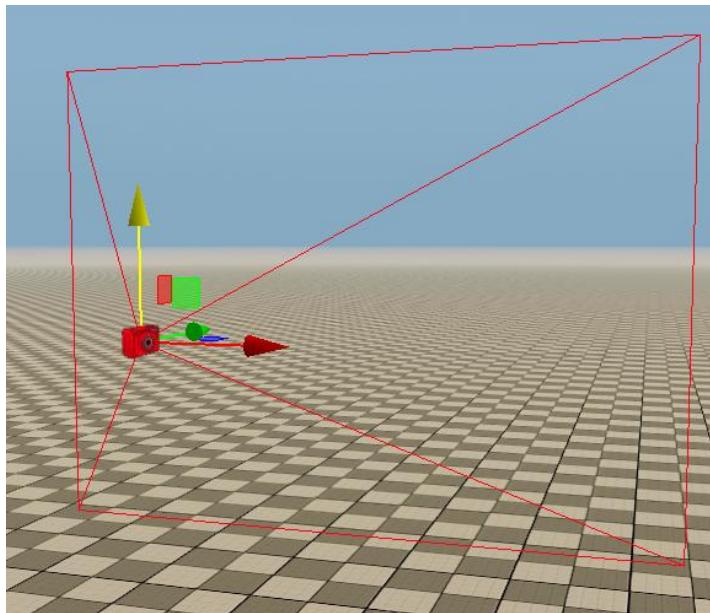


5. Save your level file to store the light values in your level.

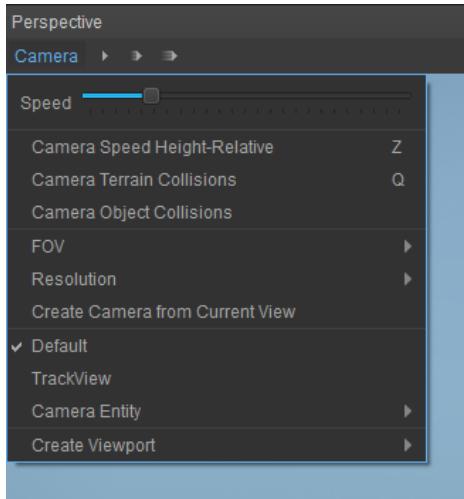
## Placing a Camera

You can place a camera in your level by adding a camera entity.

1. Open a level in the Editor, and then navigate to **Create Object -> Misc -> Camera**.
2. To add a camera, select the **Camera** entity and drag it into the Viewport. You can add multiple Camera entities to your level. You can also use the **Rotate** tool to change the viewing angle of the **Camera** entity.

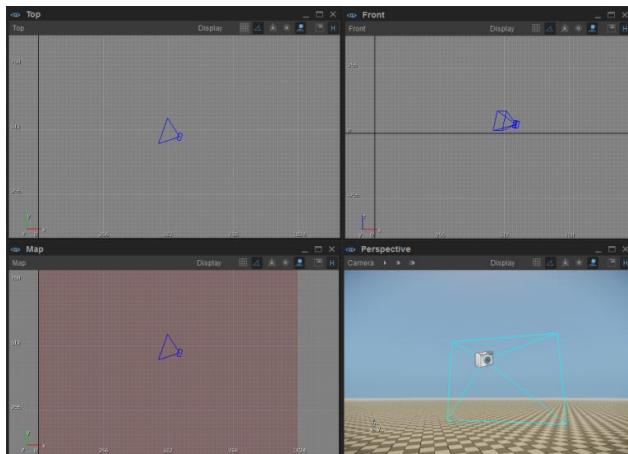


3. Specify the **Camera** settings under the Perspective viewport.



4. If you want to create a camera based on the current view in the Viewport, then adjust the view in the Editor and then select **Create Camera from Current View** option to add a camera entity automatically.
5. Use the Create Viewport option to align the camera based on the overall Viewport scene. You can choose **Front**, **Left**, **Map**, **Perspective**, and **Top** options.

# Getting Started With CRYENGINE Guide



6. Select the Default option to select the default camera view in the Viewport.

You can use the various settings which are available under the **Camera Settings** to specify different aspects such as the position of the camera, terrain collision, resolution, etc.

For more information on Camera Settings, see [here](#).

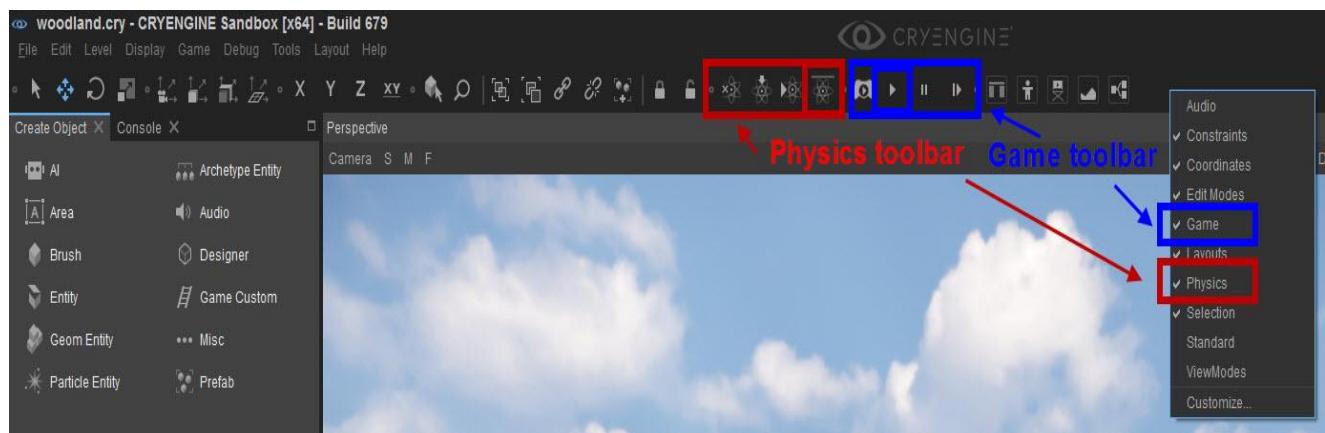
## Physics

The Physics tool is useful for testing the physics of your objects. This is an extension of the AI/Physics mode, which allows the user to interact with placed entities in the world without entering game mode.

(To use this tool, objects must be a physical entity of some sort, e.g: Basic Entity, Rigid Body, Characters, Vehicles etc.)

This tool can be found on the Physics toolbar, which you can add to the UI by right-clicking in the toolbar area of the editor.

The Physics tool is the 4th button in the highlighted red toolbar. Also, you need the Game toolbar active so you can enable AI/Physics mode.



# Getting Started With CRYENGINE Guide

---

## How to use

Once you have both toolbars active on the UI.

1. Activate the AI/Physics mode (highlighted in blue above). This puts the editor into simulation mode.
2. Press the Physics Tool button to enable manipulation mode (highlighted in red above).

Now you can select physicalized objects in the scene and interact with them.

## Controls

Using the **LMB** alone or in combination with **Ctrl** and **Shift**, we can manipulate objects in different ways. Depending on where you "grabbed" the object, this will be its pivot point (examples will be given later). Once an object has been selected, it will render the physics proxy on the object to allow you to see the physics mesh. Upon release the object will return to normal render mode.

Input	Name	Description
<b>Hold LMB</b>	Pinch	Grab the object underneath the mouse cursor for full 360 degree, no movement restrictions. The object will be completely physicalized and will be attached to the mouse until you let go of LMB.
<b>Hold Ctrl+LMB</b>	Pinch Axis Locked	Grab the object underneath the mouse cursor and your movement of the object will be locked to the X and Y planes only, in relation to the Viewport. (Its rotation is locked, but translation isn't. The object will not spin).
<b>Hold Shift+LMB (Then release)</b>	Fire Mode	This mode turns the action of the mouse cursor into a <b>projectile</b> . The longer you hold down <b>Shift</b> , the more the "impulse" of this projectile will build up. The maximum level of buildup is reached after holding down <b>Shift</b> for 3 seconds.  The buildup of "impulse" is reflected by spawning a red sphere under the mouse cursor. The longer you hold Shift and <b>LMB</b> , the bigger the red ball gets to "preview" the impulse being applied.  To "fire" the projectile, release the <b>LMB</b> and the impulse will be applied in the direction of the camera. <ul style="list-style-type: none"><li>• If you want to interact with the object gently, quickly release the <b>LMB</b> to apply a small impulse.</li></ul> Hold <b>Shift</b> until maximum impulse is enough power to send the HMMWV (2500kg) flying through the air!

When the object selected is active, it's bound to the mouse via a constraint that links them together. If you flick the object around while still holding the mouse, you'll see a thin blue line that represents the link. This link has a stretchy (rubber band) component that allows some freedom of movement at high speeds. Once deselected the link is removed. It's only temporarily created upon selection and is destroyed afterwards.

## Technical information

The following section explains how the implementation is handled, depending on the entity class you are trying to interact with.

- Pulling is handled by adding a constraint between an entity and an invisible 0-mass rigidbody (created internally), except for cloth and actors. Cloth uses cloth-specific vertex attachment and actors use a temporary 0-length rope (since they don't support constraints) with kinematic (0-mass) rigidbodies.

# Getting Started With CRYENGINE Guide

---

- For non-static objects hits (fire mode) use a general impulse interface (actors also add the same impulse to the physicalized character skeleton, if present). For static objects, a temporary fake bullet particle is created and launched. This allows it to interact with breakable vegetation.

## CVars

There are some associated CVars that control the velocity / pressure etc.. but these are for reference only. There is no need to modify these.

CVar/Command	Description
ed_PhysToolHitVelMin	Minimal velocity in insta-hit mode.
ed_PhysToolHitVelMax	Maximal velocity in insta-hit mode.
ed_PhysToolHitProjMass	Projectile mass in projectile hit mode (on statics).
ed_PhysToolHitProjVel0	Minimal projectile velocity in projectile hit mode (on statics).
ed_PhysToolHitProjVec1	Maximal projectile velocity in projectile hit mode (on statics).
ed_PhysToolHitExplR	Hit explosion radius.
ed_PhysToolHitExplPress0	Hit explosion minimal pressure.
ed_PhysToolHitExplPress1	Hit explosion maximal pressure.

## Physics Entities

Physics entities are modifiers and are used to simulate physical events such as explosions, gravity fields, and wind or to physicalize objects such as cloth, breakable entities or ropes. However, physical entities which are related to a body instead of an event need to be connected to an object in order for them to be selectable.

### To Add a Physics Entity

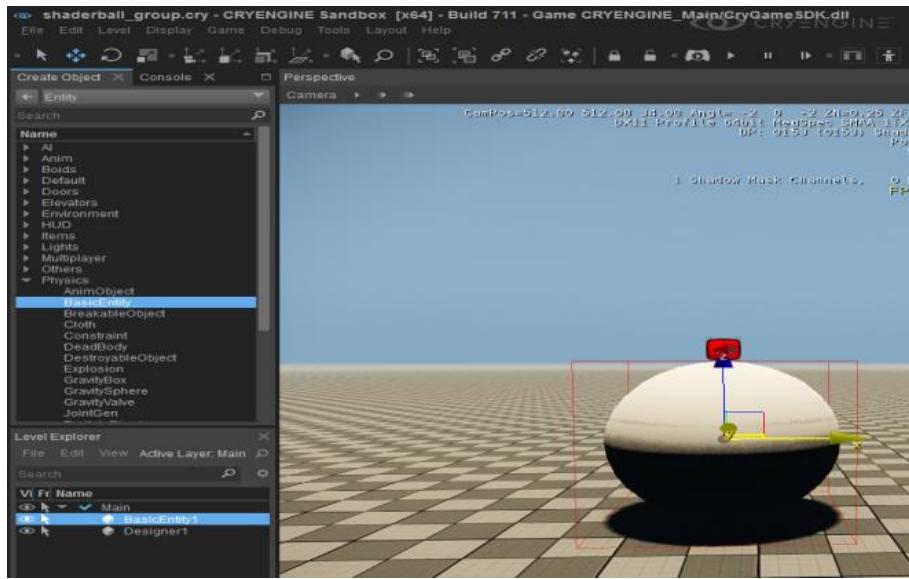
- In the **Create Objects** tab, click **Entity**.
- Under **Browser**, expand **Physics**. Drag **BasicEntity** into the **Perspective** viewport. The BasicEntity object appears as a sphere.

**Note:** You will need to first import the objects (download them from the Marketplace) into your project folder so that they can be used in a level. For example, `YOUR_PROJECT_FOLDER\Assets\Objects`

See [this page](#) to find out how to import assets to your Launcher.

# Getting Started With CRYENGINE Guide

---



3. Under the **Properties** tool, select the **Model** attribute under **Entities Properties**, then click the folder icon.
4. In **Entity Properties**, under the **Physics** section, set **Mass** to 100.
5. This turns on the physics object so that the block can respond to events. Any value above 0 turns on the physics object. Higher values indicate greater mass, which of course affects the object's behavior.

## Visual Scripting

Visual Scripting allows anyone to create gameplay systems using predefined nodes within CRYENGINE. You can use the following types of scripting in CRYENGINE.

- [FlowGraph Scripting](#)
- [Schematyc](#)

### **Flow Graph Scripting**

Flow Graph is a visual scripting system embedded within the Editor. The main advantage of the Flow Graph Editor is that users do not require any scripting or programming knowledge. Simple and complex logic can be built with just a few clicks and without requiring any knowledge of scripting or coding. A large library of nodes allows the user to fully control the entities and AI in a level.

In addition to being the main tool for creating mission logic in single-player levels, the Flow Graph can also be used to create prototype gameplay, effects, and sound design. Furthermore, levels can have multiple graphs that perform different tasks simultaneously.

Please refer to the topics below to understand the basic functionalities of the Flow Graph.

- [Creating a New Graph](#)
- [Adding and Editing Nodes](#)
- [Creating and Editing Links](#)
- [Debug Flow Graph](#)

# Getting Started With CRYENGINE Guide

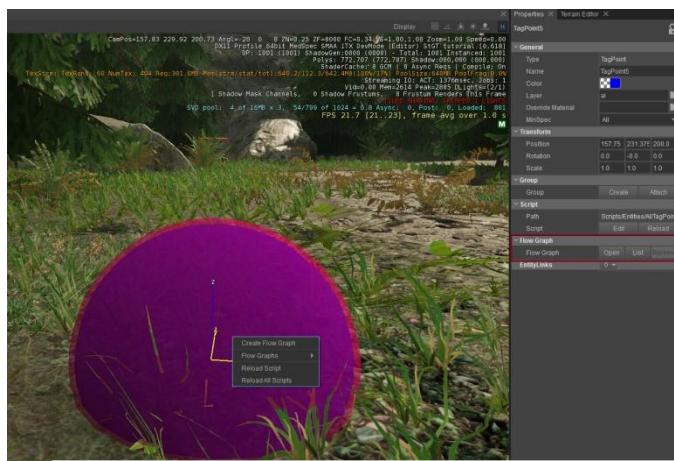
---

## **Creating a New Graph**

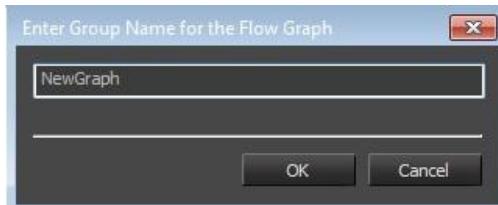
Graphs belong to a specific entity and are stored as a property of the entity. When the entity is saved or exported the corresponding graph is also automatically saved.

### **To Create a New Graph**

1. Open a level in the Editor then navigate to **Tools -> Flow Graph**.
2. Select an entity in the Viewport, and navigate to the **Flow Graph** section under the **Properties** tab. There are three buttons: **Open**, **List**, and **Remove**.
3. Click **Open** in the **Flow Graph** section under the **Properties** tab to create a new Flow Graph. You can also right-click the entity in the **Viewport**, and select **Create Flow Graph**.

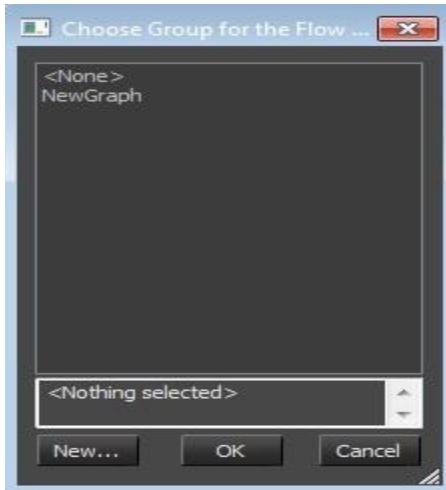


4. If this is the first Flow Graph in a level, a new **Enter Group Name for the Flow Graph** dialog box will be displayed (see picture below), this prompts you to enter a new group name for the Flow Graph.



5. Enter a name and click **OK**. This will create the new graph and automatically put it in the group.
6. If a Flow Graph group is already present, click **New** to create a new group name for the Flow Graph or select from the existing group.

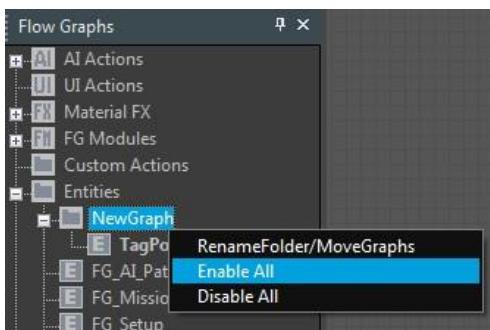
# Getting Started With CRYENGINE Guide



7. The graph overview window will show the new graph and automatically open it. You can right-click a graph node in the overview window to display a context menu with additional options to edit the graph.
8. Save the level to store the newly created flow graph.

To enable or disable the graphs, right-click the desired graph in the Flow Graph overview window and select **Disable**. By disabling, the nodes within the graphs will be ignored while the game is running. You can select the **Enable** option in the context menu to activate the graph again.

To disable an entire set of graphs within a group, right-click the group folder in the Flow Graph overview window and select **Disable All**. You can also select **Enable All** to enable all the graphs within the group.



## ***Adding and Editing Nodes***

Adding nodes to a graph can be achieved in several different ways – however it depends on whether you are adding an entity node or a component node.

- Entity nodes always operate on a specific instance of an entity in the level.
- Component nodes are independent from entities and use the entities as a target on which to perform certain actions.

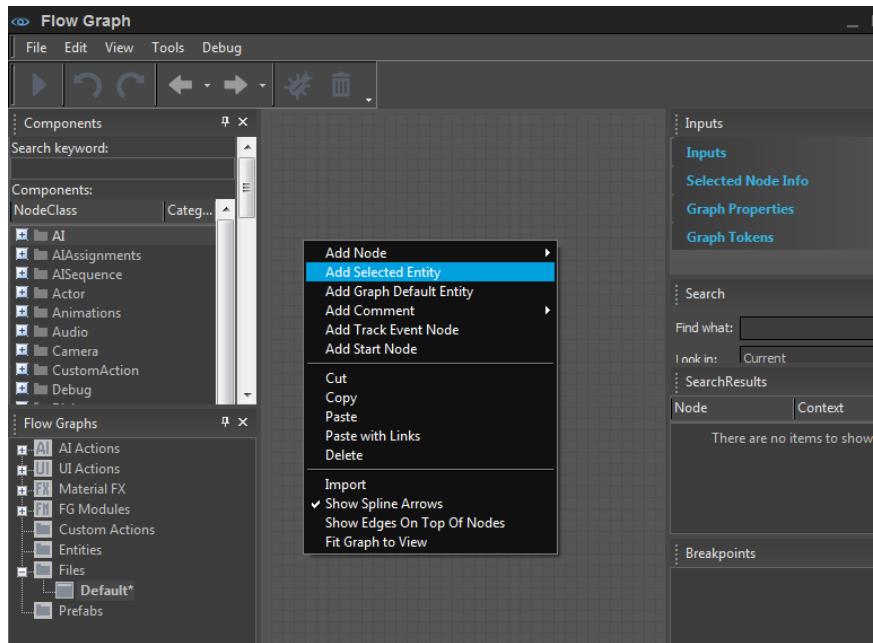
**Note:** Any entity in a level, except brushes or portals can be added to a graph.

**Note:** The target of a node can be reassigned and changed as often as is necessary.

# Getting Started With CRYENGINE Guide

## To add an Entity Node

1. Open a level in the Editor, then place and select an entity in the level.
2. Navigate to **Tools -> Flow Graph**. In the Flow Graph editor window select **File -> New**.
3. Right-click on the main graph editing pane, and then select **Add Selected Entity**.



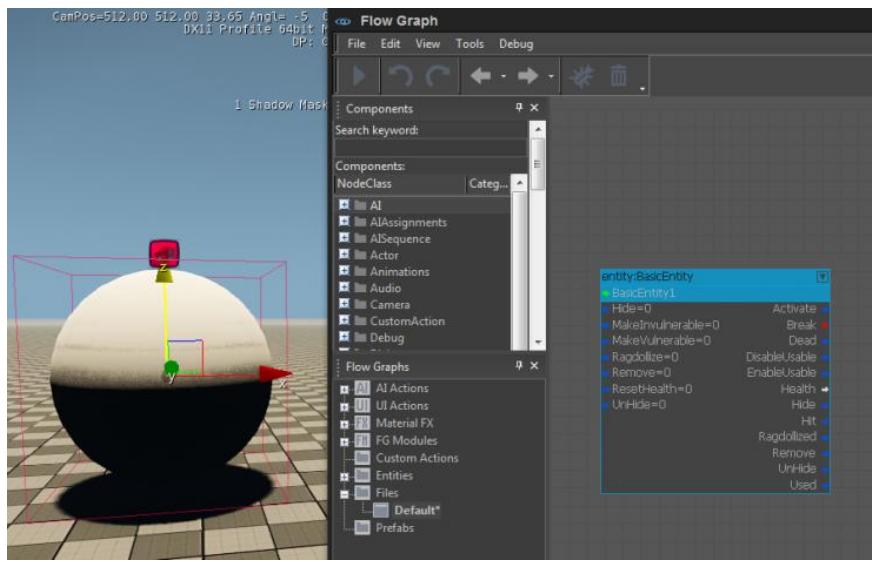
**Note:** This procedure only works if an entity has been selected in the Perspective viewport. The context menu option **Add Graph Default Entity** adds the entity's node that has already been created in the Flow Graph.

4. Since you have selected the previously created Physics entity in the Viewport, your Flow Graph should contain the following view.

**Note:** You will need to first import the objects (download them from the Marketplace) into your project folder so that they can be used in a level. For example, `YOUR_PROJECT_FOLDER\Assets\Objects`

See [this page](#) to find out how to import assets to your Launcher.

# Getting Started With CRYENGINE Guide



## ***Adding a Component Node***

Component nodes can be added within the graph and do not require any selected entity. There are three ways to add these nodes.

- Right-click the context menu on the graph edit window.
- In the Components menu you can use the Search keyword window to locate your component.
- QuickSearchNode (Shortcut: Q).

### **To Add a Component Node**

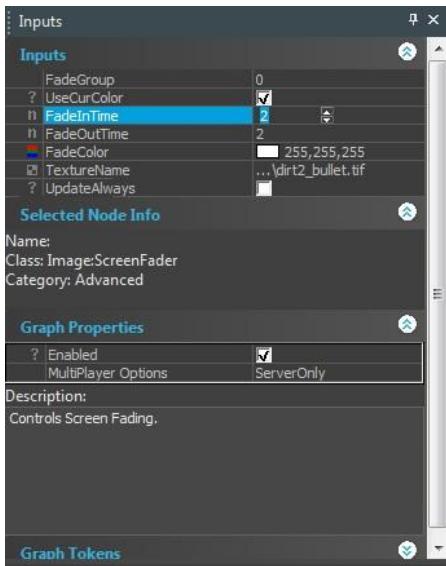
1. Right-click on the main editing pane, and then select **Add Node**.
2. Select a node from a list of sub-folders.
3. Select Entity to open the folder with the entity-related component nodes, and then select any node.

## ***Editing Nodes***

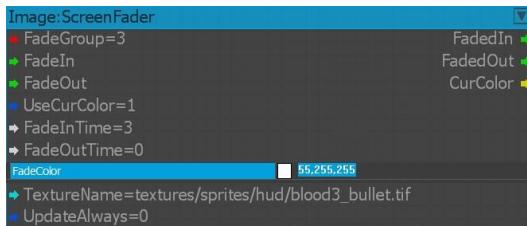
There are two ways of editing node properties.

- Navigate to the Node window located on the right hand side of the **Flow Graph Editor**, select the node and then under the **Inputs** tab modify the parameters.

# Getting Started With CRYENGINE Guide



- Double-click the parameters under the Node. This allows you to change any of the user-variable properties directly at the Node location



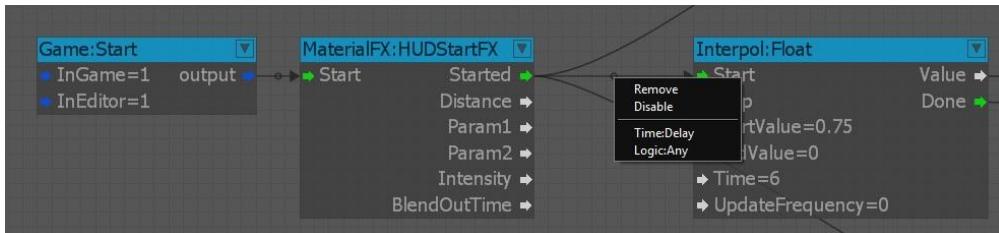
**Note:** Make sure you zoom-in on the particular node before you edit using the double-click option.

## Creating and Editing Links

To create a new link between two nodes, click any output port and drag it to the desired input port. When the nodes are moved the link will automatically adjust itself.

To delete a link, right-click on the link, and then select **Remove**. You can also click the input port and drag the link away from the port. When the link is dropped on the background pane the link will be removed.

Every link has a context menu that can be opened by right-clicking the small dot in the middle of the link.



The link context menu has four entries. The **Remove** option removes the selected link and the **Disable** option disables the selected link. A disabled link is grayed out, and is not processed - the output port is activated.

If a link has been disabled you can use the **Enable** option to enable the link again. A **Time:Delay** node provides a delay between the nodes connected by the link with a default delay time of 1 second. Note that each input port

# Getting Started With CRYENGINE Guide

---

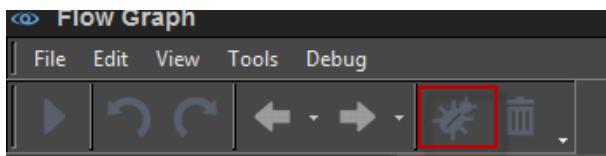
can only have one link connected to it. If you want to use more links on a single input port, Logic:Any acts as a helper node. Output ports are not limited and can have an unlimited number of links.

## **Debug Flow Graph**

If your script does not work as you expect, you can turn on **Debug** in the **Flow Graph** editor. Debug is a helpful tool to detect where your logic has failed.

### To Turn on Debug

1. In the **Flow Graph** editor's toolbar, click **Debug**.



2. Move the flow graph window to the side (but still have it visible).
3. Run the level (**Ctrl+G**). You will notice how each action in the level is displayed in the flow graph events.
4. Exit the level. Click the **Trash** icon (next to the **Debug** icon) to clear the debug events.
5. Save your file.

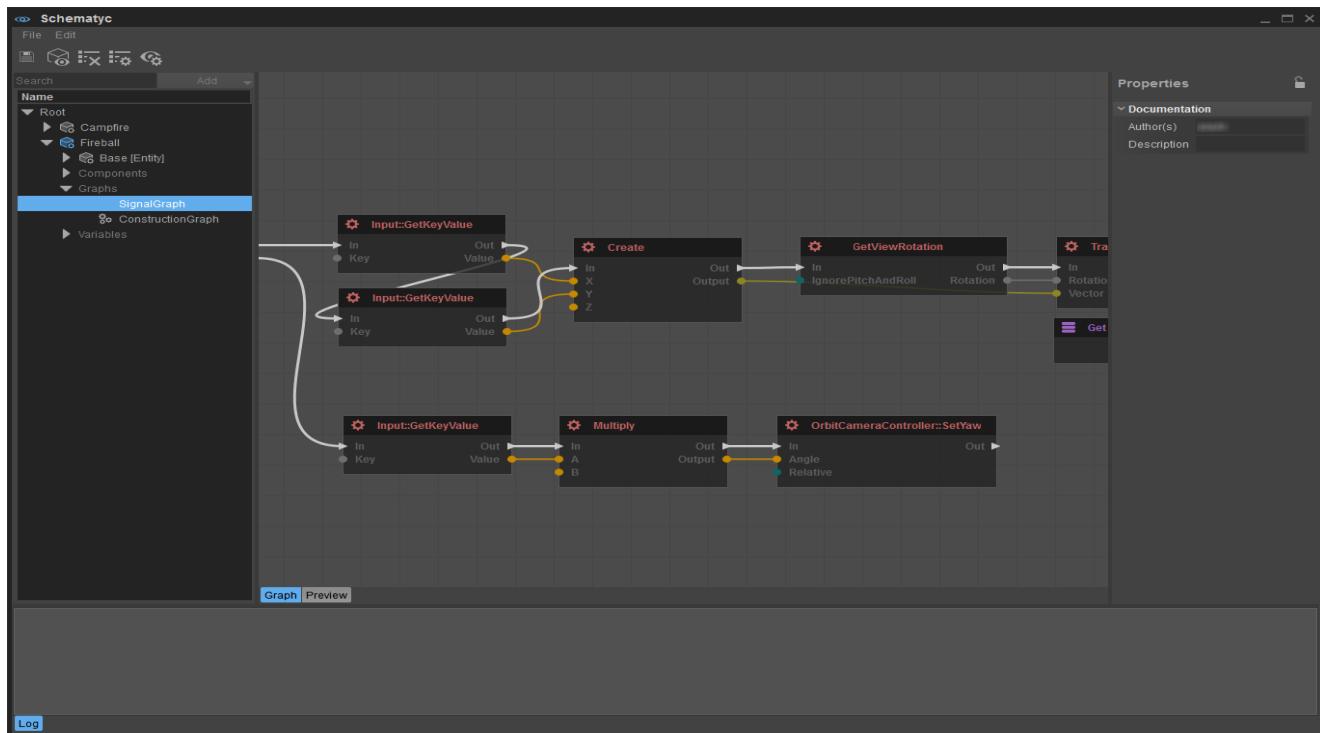
# Getting Started With CRYENGINE Guide

## Schematyc

Schematyc provides designers the power to construct new and reusable objects from a set of building blocks provided by programmers. At first glance, the editor looks a little like Flow Graph, but the two systems were built with very different purposes in mind. Whereas Flow Graphs are great for level scripting, Schematyc is designed to provide more finite control of the objects within those levels.

Visual scripting tools let you script your own logic. State machines help to break up and structure logic, making objects easier to debug and simplifying the process of network synchronization. Determinism and reduced latency make it possible to take new gameplay systems beyond the prototyping stage without the need to re-write them in C++.

Context is king in Schematyc and our aim is to only present users with relevant information, keeping the UI from becoming too cluttered and overwhelming.



## Basic Concepts

The basic concepts you need to understand before using Schematyc are.

- **Classes** - [https://en.wikipedia.org/wiki/Class\\_\(computer\\_programming\)](https://en.wikipedia.org/wiki/Class_(computer_programming)) - A class describes an object and encompasses all of the data and logic associated with that object.
- **Components** - Components are the building blocks that define what functionality is available to an object. If you want to load geometry you will need to add a geometry component to your object, to read input from a controller you must first add an input component, and so on.
- **Variables** - [https://en.wikipedia.org/wiki/Variable\\_\(computer\\_science\)](https://en.wikipedia.org/wiki/Variable_(computer_science)) - Variables store information and exist for as long as the object or state they belong to exists. By making a variable public you allow other systems access to that variable e.g. when you place a Schematyc entity in a level you can edit its public variables in the main Properties panel.

# Getting Started With CRYENGINE Guide

---

- **Finite State Machines** - [http://en.wikipedia.org/wiki/Finite-state\\_machine](http://en.wikipedia.org/wiki/Finite-state_machine) - In Schematyc, states can be structured hierarchically and all the children of a particular state inherit behavior from that state, allowing them to share both functionality and transitions.
- **Signals** - Signals are used to communicate between objects and object states. For example, you could create a damage signal to send damage from one object to another. Signals can also be used to send data, so in the example of the damage signal you might add two variables, damage type and damage amount, in order to specify the damage being dealt.
- **Functions** - <https://en.wikipedia.org/wiki/Subroutine> - Functions in Schematyc are just like functions in any other programming language; you pass inputs, the function performs some logic, then you return the output.
- **Enumerations** - If you want to list a finite set of options with string identifiers you use an enumeration.

For a series of videos about Schematic, follow [this link](#).

## Particle System

With CRYENGINE V we introduced Wavicle (Particle Editor 2). This is a new particle system that has been designed to offer more flexibility to the designer and to make optimal use of modern CPU and memory architectures. To achieve this goal, the particle simulation backend implements a data-oriented design, where memory layouts have been optimized for vectorized processing.

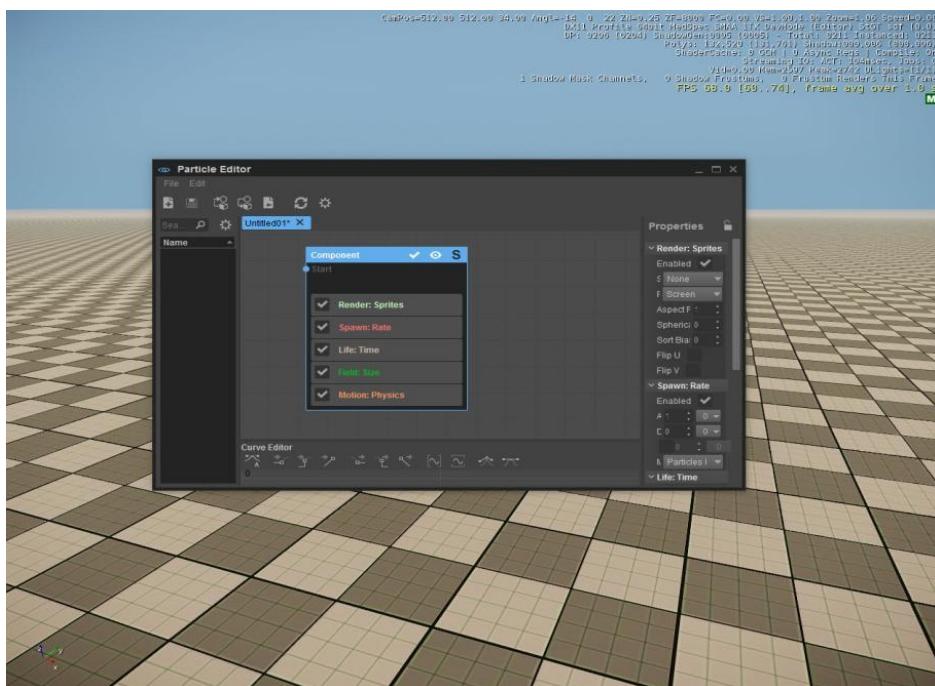
At the frontend side, the particle system presents a modular setup approach that allows for the flexible design of particle effects that can be setup to interact with the rest of the Engine though attribute passing. For more information on the Wavicle Particle System please see [this page](#).

### ***Creating and Adding a Particle Effect in Your Level***

Go to Tools -> Particle Editor. Particle Editor 2 uses a node based system that helps to better visualize particle dependencies.

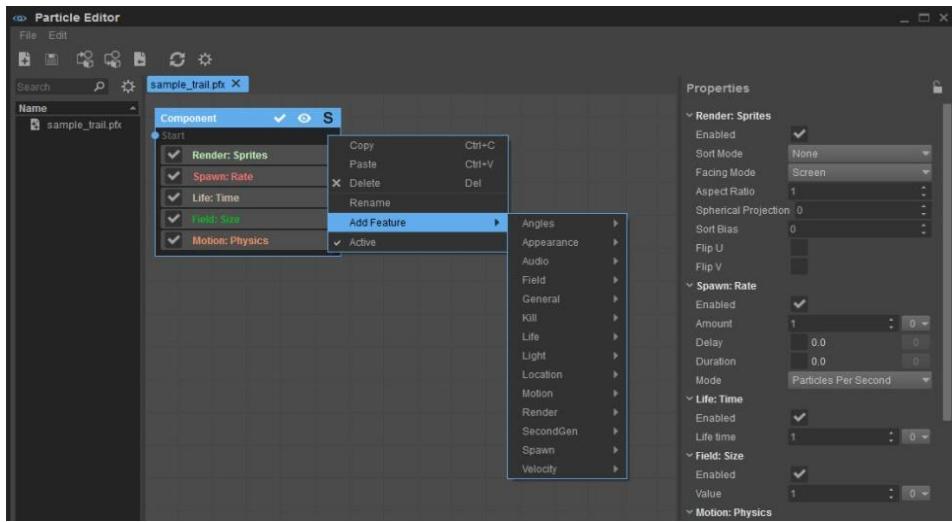
#### **To Create a Particle Effect**

1. In the Particle Editor window, click the **New Effect** button  to load a new effect.



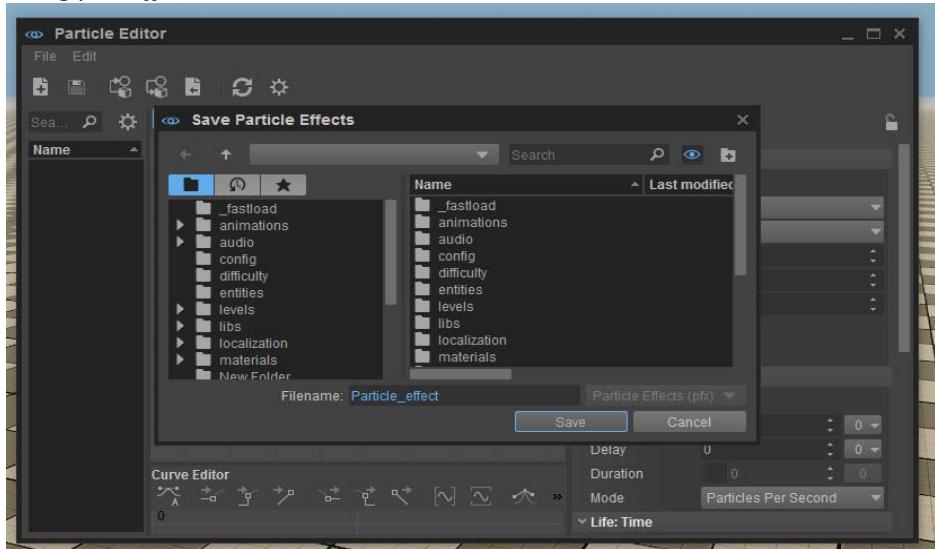
2. Add new features to the component in the **Particle Editor** as shown in the image below, or you can edit the existing features using the **Properties** tab in the **Particle Editor**. For more information on the Particle Editor UI options, please see [this page](#).

# Getting Started With CRYENGINE Guide

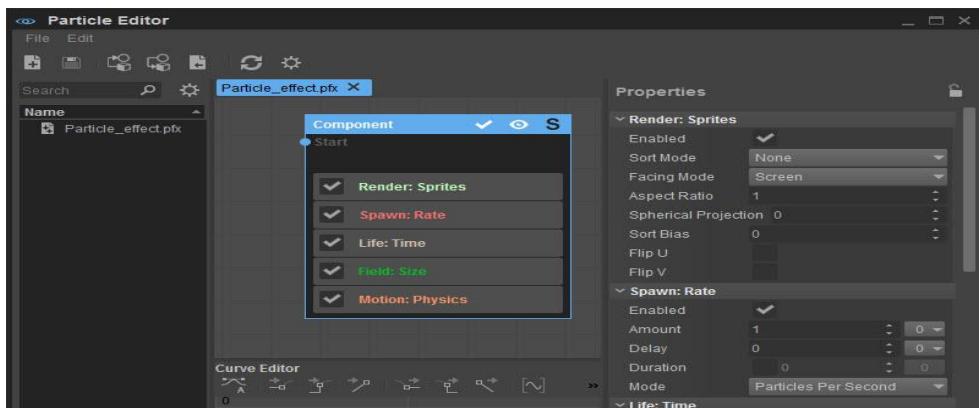


- Click the **Save** button and enter a preferred filename to save your effect in the project directory.

*Saving your Effect in the Particle Editor*



- Now you have successfully created and saved an effect using the **Particle Editor**.



# Getting Started With CRYENGINE Guide

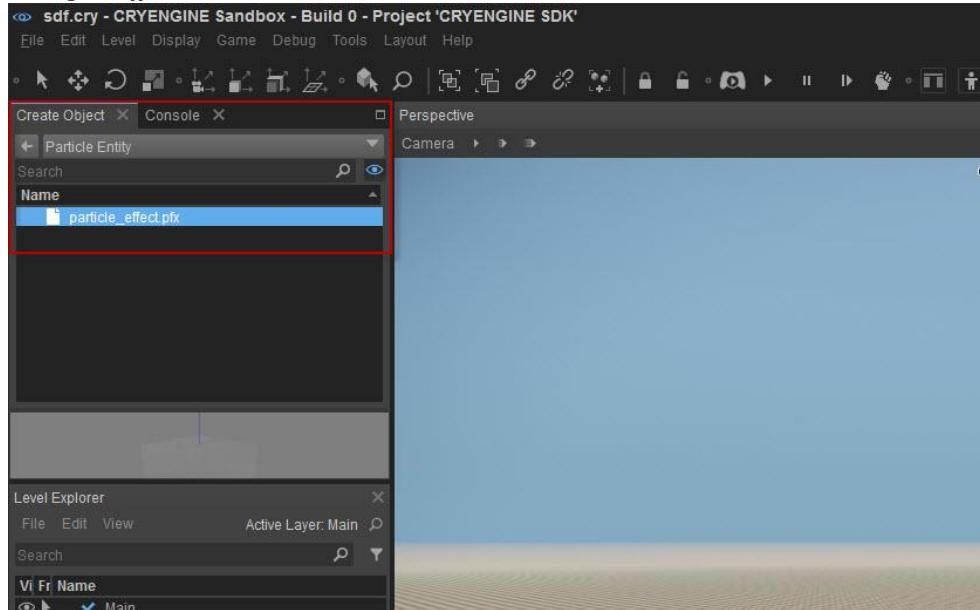
---

## To Add an Effect into Your Level

Once you have created an effect in the Particle Editor you now need to import the effect into your level in the Editor.

1. Open a level in the Editor.
2. Under the **Create Object** tab, click the **Particle Entity** option to select the particle effect.

### *Adding an Effect into Your Level*



3. Now you can drag the entity to be placed in your level. You can also drag the effect directly from the Particle Editor.
4. You can select an object in your level and then click the **Apply to Selected Entity** button to apply the effect to the selected object in your level.

## Chapter 8: Using GitHub for CRYENGINE

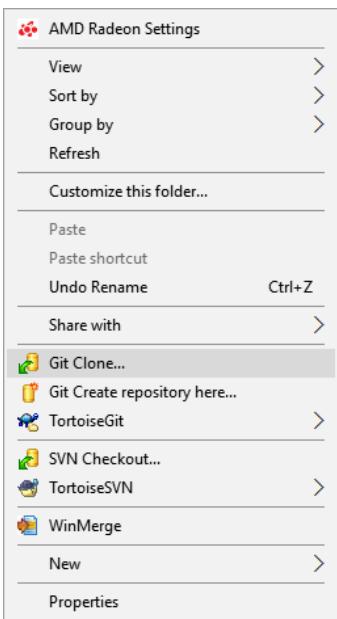
As of CRYENGINE release 5.1.0, Engine source code has been distributed via repositories hosted at <https://github.com/CRYTEK/CRYENGINE>.

### Configuring a Git Client

You can use different Git clients to access CRYENGINE source code, but we recommend using TortoiseGit which provides a simple interface.

#### To Configure a Git Client

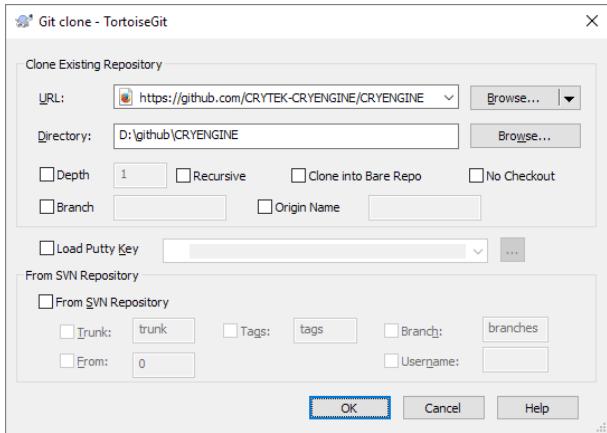
1. Install Git for windows from <https://www.git-scm.com>
2. Install TortoiseGit from [www.tortoisegit.net](http://www.tortoisegit.net)
3. Open the directory in which you want to clone a repository, and right-click and select **Git clone** from the context menu.



4. Enter the following details in the Git Clone window.
  - a. <https://github.com/CRYTEK/CRYENGINE/CRYENGINE.git>
  - b. [Enter git credentials if prompted]

# Getting Started With CRYENGINE Guide

---



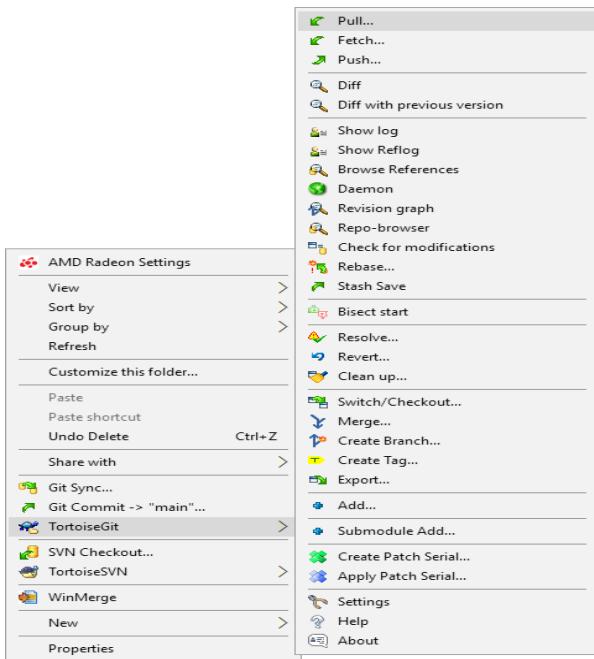
5. Click **OK**, and wait until the files and history are downloaded to your local machine.
6. Download the CRYENGINE SDKs from <https://github.com/CRYTEK-CRYENGINE/> and unzip them into <repo root>/Code/SDKs.
7. Follow the instructions on the [Getting Started with WAF](#) page to build the code. The presence or absence of SDKs will be detected automatically.

## Updating Your Repository

You will need to update your repository to reflect the latest source files.

### To Update Your Source Code

1. Right click in the git repository.
2. Select **TortoiseGit -> Pull**.



# Getting Started With CRYENGINE Guide

---

3. If you have made changes, it is recommended to run **TortoiseGit -> Fetch** and then merge your local changes.
4. If you are updating from one release to another be sure to download the corresponding SDKs.

## **Restricted Access Source Code**

If you have a license for CRYENGINE that includes the Sandbox Editor source code, please let us know your Github username so that we can grant you access to the appropriate repositories.

Similarly, in order to get console repositories you need to be a Sony PlayStation or Microsoft Xbox verified developer. Please send your verification documents and Github username to [contact@cryengine.com](mailto:contact@cryengine.com). Once approved by us you will be notified and activated to download the consoles packages.

## Chapter 9: Glossary/Terminology

Terms	Description
<b>Actor</b>	A specialized entity that is the basis for characters in a game.
<b>Additive Animation</b>	An animation that can be attached to a base animation to extend its behavior.
<b>Alpha Channel</b>	An extension of RGB color values for specifying the opacity of an object. A value of 0.0 indicates fully transparent while a value of 1.0 indicates fully opaque.
<b>Archetype Entity</b>	A special type of entity with linked instances. If a parameter of the archetype entity is changed, all other instances of that entity parameter are automatically updated.
<b>Asset</b>	Any art, texture, 3D model, sound effect, or other digital data that is presented to the user in the game.
<b>Attachment</b>	A hierarchical object that is attached to characters, responds to real-world physics, and can be attached, detached, or replaced at runtime in the game. Character attachments include clothing, weapons, tools, or entire body parts such as heads or hands.
<b>Blend Shape</b>	Method that stores a deformed version of a mesh as a series of vertex positions. In each keyframe of an animation the vertices are interpolated between these stored positions.
<b>Blend space</b>	Animation blending that is treated as geometry. A character's kinematic, physical, and other high-level motion-related parameters are mapped onto corresponding features that are stored in animation clips. By storing such motion as parameters, controllable interactive animations are possible. Specifically, an animation is associated with a 1D, 2D, or 3D location in the blend space. Also known as a <i>bspace</i> .
<b>Boids</b>	Entities that mimic living animals and that have simulated group behavior and obstacle avoidance.
<b>Brush</b>	A simple 3D shape that is tied to an entity, and that provides a specific appearance. Brushes are used for static objects.
<b>Collision Proxy</b>	A simplified geometric shape for approximating a more complex piece of geometry for purposes of fast first-pass collision detection.
<b>Cubemap</b>	A set of six squares that represent reflections from the environment. The six squares form the faces of an imaginary cube that surrounds an object.
<b>Cutscene</b>	A non-interactive cinematic game sequence that is typically used to promote plot during gameplay.
<b>DCC</b>	Digital Content Creation; related to a third-party product such as Autodesk 3ds Max or Autodesk Maya and used for creating digital assets.
<b>Decal</b>	A 2D texture placed on a piece of flat geometry.
<b>Detail Map</b>	An image for adding up-close surface details to an object.
<b>Diffuse Map</b>	An image for defining the base color and pattern of an object's surface.
<b>Displacement Map</b>	A type of height map that modifies the position of vertices of a surface by a specified amount.
<b>DOF</b>	Depth of Field. The degree to which distant objects are in focus (relative to closer ones).
<b>Emitter</b>	An entity that specifies the location from which particles are emitted.

# Getting Started With CRYENGINE Guide

---

Terms	Description
<b>Entity</b>	A game object with one or more components that provide some behavior or functionality. An entity consists of a unique ID and a container.
<b>Environment Probe</b>	A technique that uses cube maps to provide a game level or location with realistic ambient lighting.
<b>Game Object</b>	A game object is any object in a game that the player can see and/or can interact with. The player object, power ups, enemies, platforms, walls, weapons (if collision detection is considered) and projectiles are all game objects
<b>Gloss Map</b>	An image that represents the micro-scale roughness of a surface. The gloss map is located in the alpha channel of the normal map.
<b>Heightmap</b>	A grayscale image used to modify vertex positions of a surface. CRYENGINE uses heightmaps to store terrain surface height data. White areas represent the high areas while black areas represent the low areas of the terrain.
<b>Helper</b>	Visual icons attached to objects in the Sandbox Editor and that provide object-specific functionality.
<b>Keyframe</b>	An animation frame that specifies exact positions and orientations of geometry affected by the animation. Animation frames that exist between keyframes are interpolated based on animation curves.
<b>Level</b>	A world or map that represents the space or area available to the player during the course of completing a discrete game objective. Most games consist of multiple levels.
<b>LOD</b>	Level of Detail. A technique for increasing performance and reducing draw calls by displaying progressively less-detailed objects the farther they are from the camera.
<b>Mesh</b>	A collection of vertices that define the surface of an object.
<b>Minimap</b>	A miniature map placed at a screen corner in the game to aid players in orienting themselves in the world.
<b>Morph Target</b>	A snapshot of vertex locations for a specific mesh that have been deformed in some way.
<b>Normal</b>	The vector that is orthogonal to a surface defined by a set of vertices.
<b>Normal Map</b>	An image whose pixel values are interpreted as the normal vectors for each point on the surface to which the image is mapped.
<b>Parallax Mapping</b>	A technique that is used to create detail in a texture adding the illusion of depth. This depth perception changes based on perspective.
<b>PBR</b>	Physically Based Rendering. PBR uses real-world physical rules and properties to define how light interacts with the surface of objects. Used by the CRYENGINE rendering system.
<b>POM</b>	ParallaxOcclusion Mapping. POM uses a displacement map to encode surface detail information in a texture. In this way self-occlusion and self-shadowing of an object is possible without changing the surface geometry.
<b>Prefab</b>	A game object template that stores an asset or a group of assets and all associated properties.
<b>Procedural Vegetation</b>	A technique used to automatically cover a large area of terrain with vegetation objects using texture layers.
<b>Project</b>	The collection of levels, assets, and code that make up a game.
<b>Retargeting</b>	Applying animations that were created for one model to another.
<b>Rigging</b>	The process of building skeleton hierarchies of bone joints for a character mesh.

---

# Getting Started With CRYENGINE Guide

---

Terms	Description
<b>Rope</b>	Used for attaching cloth, hair, or ropes to a character so that the objects can dangle and move realistically against the character.
<b>Shadow Map</b>	A technique for controlling how shadows are added to a scene. You can use multiple cascaded shadow maps to control how sun shadows look at varying distances.
<b>Skinning</b>	The process of binding bone joints to a model's mesh (skin).
<b>Socket</b>	A pivot point on a character where attachments are connected. Attachments dangle or move according to the properties of the socket.
<b>Specular Map</b>	An image that determines the shininess of each area of a surface.
<b>Sprite</b>	A 2D bitmap image. Multiple sprites can be grouped into a single image known as a sprite sheet.
<b>Tessellation</b>	The deformation of a surface using one or more geometric objects with no overlaps or gaps. Tessellation increases the geometry count of the mesh by subdividing polygons into smaller polygons before it gets displaced.
<b>Texture Mapping</b>	The application of an image to a surface.
<b>TOD</b>	Time of Day. TOD is used to simulate the changing lighting conditions as the sun crosses the sky in a level.
<b>UV Mapping</b>	The projection of texture coordinates onto a 3D surface.
<b>Vertex Color</b>	A method for adding variety, depth, and color variations to an object surface.
<b>Voxel</b>	A volumetric point in a 3D space, similar to a pixel in a 2D space.