360° Video in Unity

Equirectangular footage is cool, but viewing it in VR is cooler....

While Unity is a game engine and designed for making video games, it's use cases go much further than that. You can use Unity to create a standalone executable that plays your 360° masterpiece or a *Bandersnatch* style choose-your-own 360° adventure.

Preparing the 360° Video

Unity is resolution agnostic meaning it can accept whatever obscure video resolution you throw at it. That said, just because it can accept it does not mean it will look good. For this guide we are using 2:1 equirectangular footage as that works best but you could also use a cubemap projection or an over/under stereoscopic projection with some minor modifications.

It's also important to keep in mind that higher resolutions are more CPU intensive on a computer that is already driving a VR rig. You should choose an export resolution that is matched with your hardware. For example, an HTC Vive can't show 8K video so it would be a better use of resources to downscale 8K footage to 5K.

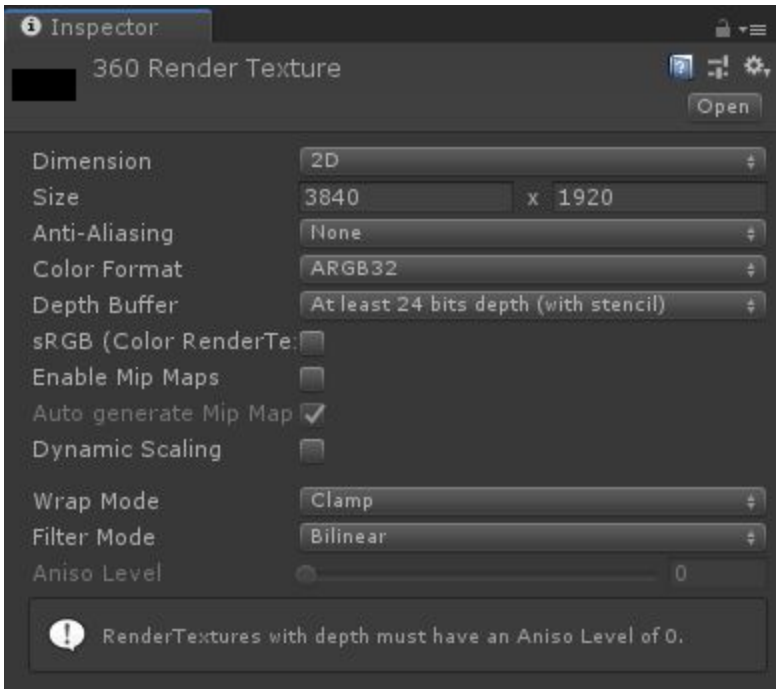
Recommended Settings

- **Encoding:** H.264
- **Format:** .mp4
- **Resolution:** 3840 x 1920

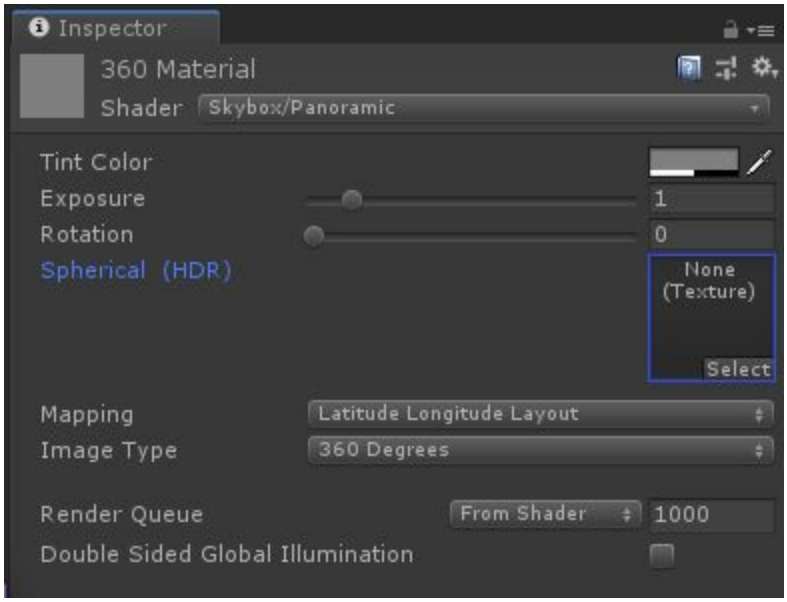
Creating a Unity Video Player Scene

With your video prepared you'll just need to create a new Unity scene to contain it in. This guide assumes that you've just opened a brand new Unity project.

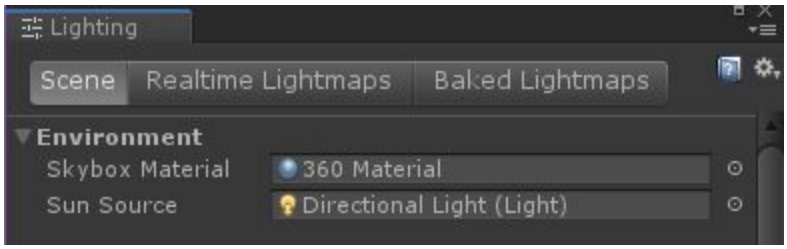
1. Import your 360° by selecting Assets ➔ Import New Asset... from the menu bar and navigating to your video file.
2. Create a new Render Texture by selecting Assets ➔ Create ➔ Render Texture from the menu bar.
    a. A render texture is a special type of texture that updates at runtime and in our case will server as a destination for the video as it plays.
3. Rename the Render Texture to something that identifies it, I chose "360 Render Texture."
4. Select the 360 Render Texture and in the Inspector panel, change it's resolution to match your video's resolution.



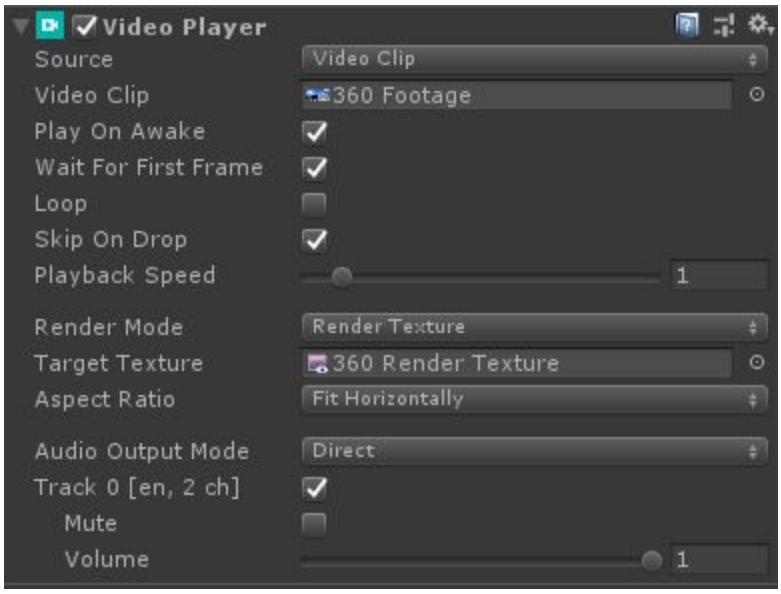
5. Create a new Material by selecting Assets ➔ Create ➔ Material from the menu bar.
    a. Materials define how a surface will be rendered based on the textures assigned to it. It will use the 360 Render Texture in this case.
6. Rename the Material to something that identifies it, I chose “360 Material.”
7. Select the 360 Material and in the Inspector panel, select “Skybox ➔ Panoramic” from the Shader property.
8. Drag the 360 Render Texture from the Project panel into the “Spherical (HDR)” property.



9. Select Window ➔ Rendering ➔ Lighting Settings from the menu bar.
10. Replace the “Default-Skybox” by dragging the 360 Material from the Project panel into the “Skybox Material” property.
    a. This makes the 360 video act as the all encompassing environment of the scene.



11. Create a new Game Object by selecting GameObject ➔ Create Empty from the menu bar.
12. In the Hierarchy panel, rename the newly created GameObject to something that identifies it, I chose “360 Game Object.”
13. With the 360 Game Object selected, choose Component ➔ Video ➔ Video Player from the menu bar.
14. In the Inspector panel, drag the video footage from the Project panel into the “Video Clip” property and the 360 Render Texture into the “Target Texture” property.
    a. This caused the render texture to server as the “screen” for the video within the scene.

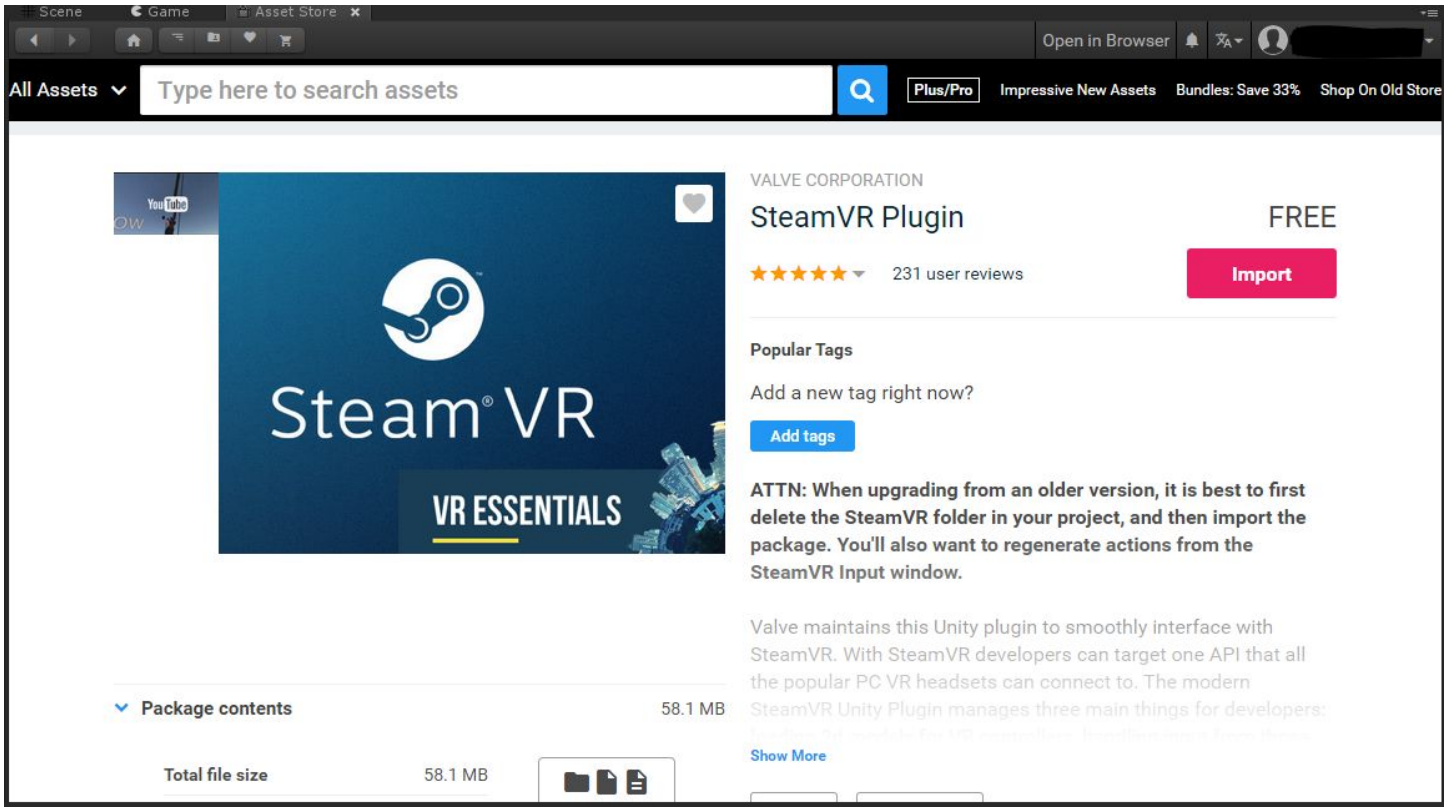


At this point the video is now the environment which surrounds your scene. You could stop here if you are trying to use your footage as a custom skybox, or you can continue to create a fully realized VR player.

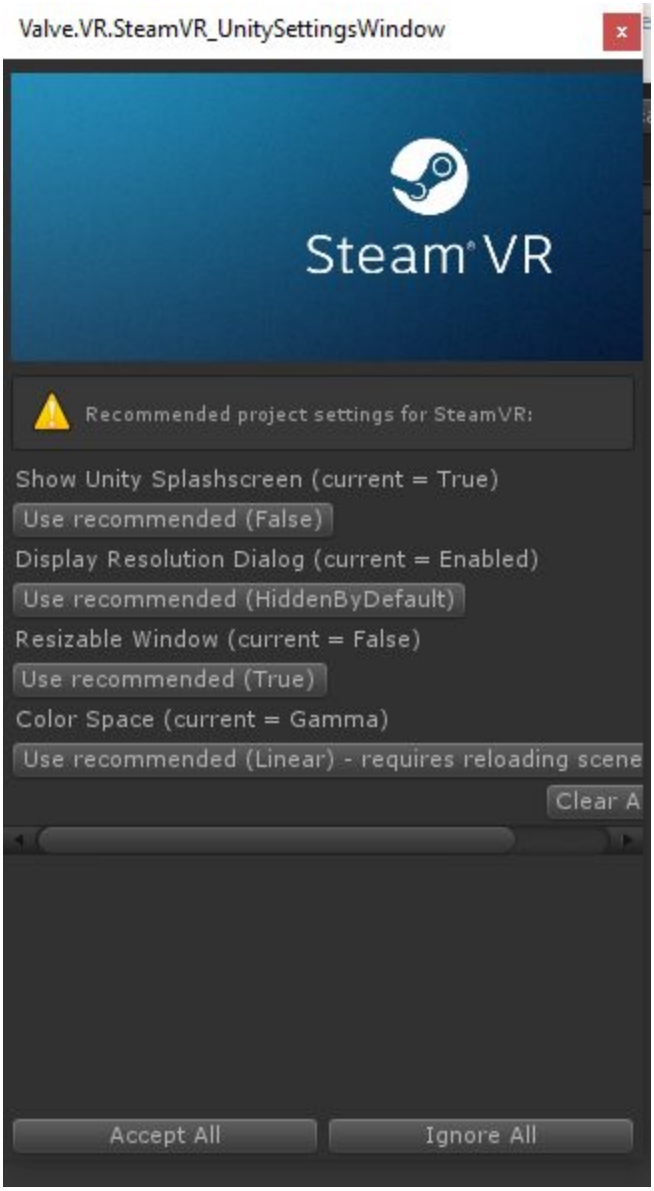
Putting a 360° Scene into VR

With the scene set up with a 360° environment, it is not too difficult to transform it into a VR scene.

1. From the Asset store panel, search for, download, and import the “SteamVR Plugin” offered by Valve Corporation. It will take some time to download and subsequently import.



2. SteamVR as a tendency to throw popups about calibration settings. Choose “Accept All” should it pop-up.



3. SteamVR will have added itself as a folder in the Project panel. Inside that folder navigate to "Prefabs" and drag "[Camera Rig]" into the Hierarchy panel.
4. In the Hierarchy panel remove the "Main Camera" asset.
5. The first time you hit play with SteamVR in the scene it will ask to create input bindings, select "Yes" and "Save and Generate."

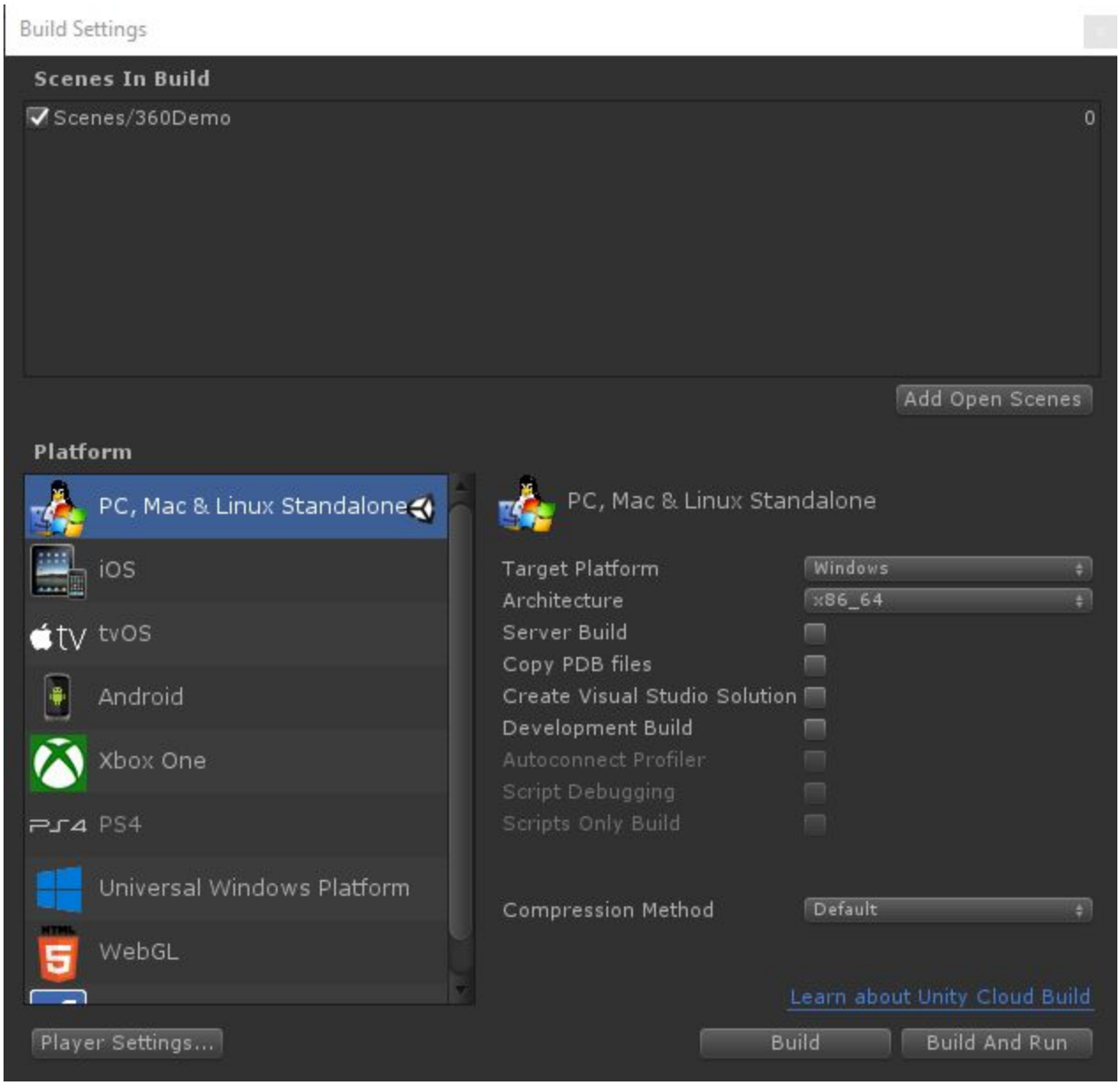
You now have a VR-ready 360° video player.

Building the VR 360° Video Player

In Unity, building is the process of compiling assets into a standalone executable file. We're going to do this to make a standalone 360° Video Player application.

1. Save the scene by selecting File ➔ Save from the menu bar.
2. Open the Build Settings by selecting File ➔ Build Settings from the menu bar.
3. In the resulting window, choose "Add Open Scenes."
4. The "Player Settings" will allow you to change the application's name, icon, author and other metadata.
5. Ensure the "Target Platform" is Windows as our VR systems only run Windows.

6. Click “Build” and choose where you want to save your build, preferably in a new empty folder.



You now have a completed standalone executable which contains your video. Note that Windows builds include a sidecar folder called “Data” and several accessory files so it is best to move it within the containing folder to other devices.

Next Steps

From this project you could build a UI interface inside Unity, add interactable objects, etc. The possibilities are endless.