

**THANK YOU FOR PURCHASING**  
**VRSI: VIRTUAL REALITY STEREO IMPOSTERS**



## TABLE OF CONTENTS

Introduction

What is VRSI?

What is included in VRSI?

Quick Start Guide

Step 1

Step 2

You're Now Ready to Create a VRSI!

Multi-View VRSI Creation

FAQ

Which render path is VRSI Compatible with?

Is GPU instancing supported?

Do you have any recommendations?

Support Contact

Change Log

## Introduction

This is a Game Changer for Unity VR! Achieve Impossibly Complex and Dense Scenes even on very limited standalone VR headsets such as the Oculus Quest 1/2. The generated VRSI's look indistinguishable from their real 3D mesh counterparts. They can be dynamically lit by any light source, cast and receive shadows, animate (such as wind animation on trees and other vegetation) just like the actual object after a very simple and rapid creation process in VRSI Creation Studio.

*“Please note: To truly appreciate the Power and versatility of VRSI, it must be experienced inside Virtual Reality”*

## NEW! 360° degree Multi-View VRSI

Creation option has now been added with V1.2 These allow you to generate view dependent imposters that change depending on where they are being viewed from, providing a much better sense of an actual 3D object.

## As of V1.2 VRSI can now also be used for Standard PC and Mobile Projects!

With the introduction of V1.2, VRSI has complete support for both Non-VR PC and Mobile projects . Relevant and highly optimized shaders are included and it is simply a one click process during the creation process. Take advantage of all the power and ease of use of VRSI to add extremely high quality relightable billboards into your project now.

## What is VRSI?

**Traditional** 2D Billboards or Standard Stereo Pair Images in virtual reality cannot be effectively used as an imposter to show a realistic representation. It either looks flat or suffers from the known stereo image pop-out effect as

viewing distance changes. VRSI provides True 3D Stereo Billboards that look almost identical to the real 3D mesh object, even better most of the time.

It is important to understand, VRSI is the only method to generate and display 3D Imposter billboards in VR. Standard Side-By-Side stereo pair texture 3D billboards cannot be used in VR because of *locked image separation*. They will exhibit extreme Pop-Out when the viewer is far away and it will not fit in with any other existing object.

VRSI solves these issues by taking advantage of a feature only available in Virtual Reality. VRSI allows for an ultra-realistic representation that can be used as a true 3D Imposter, using multiple or even thousands or millions of instances, as there is almost zero performance impact! Please keep in mind the generated VRSI billboards are best seen inside VR as the user will see the complete depth parallax.

- Achieving Infinite Detail and Complexity in VR with True Stereoscopic 3D Imposters at almost Zero Performance Impact is now Possible. They look great at 'Any' Distance, even at extreme close-ups.
- Exclusively developed to work with VR, VRSI will allow everyone to populate their VR worlds/scenes to look incredibly dense, a 5000000 Polygon Tree can be represented with a single polygon with full depth stereo parallax inside VR! This is Indispensable for any VR project when performance matters.
- When you want to achieve the highest level of detail, density and Complexity at the Highest Performance, VRSI is the ideal tool for your arsenal.

By utilising our advanced custom renderer and process, any object can be converted to a VRSI in seconds regardless of the shaders they use. The generated billboards are of extremely high quality with multiple options, including Alpha-to-Coverage.

## ***What is included in VRSI?***

VRSI is a Unity tool and there are no other dependencies. Once a *VRSI* is created (i.e. *The Correct stereo pair texture map with its corresponding normal map is generated inside VRSI Creation studio*) in the VRSI Studio scene, two prefabs are created, Simple (Static) and Ylock (Camera Facing) and the default shaders are attached ready to be used.

Depending on your requirement, you can change the attached Shader of the VRSI prefab by selecting the prefab and choosing the required shader from the VRSI shaders list.

*VRSI* includes the following shaders in multiple variants, *Camera Facing*, *Static*, Multi-View, PC & Mobile.

**Camera Facing:** The billboard will always face the camera, suitable for trees, foliage, grass etc. and it is locked in the Y-axis for VR compatibility.

**Static:** The billboard will not face the camera and can be scaled/moved/rotated to the desired location. Suitable for rocks, props, etc.

**Multi-View:** Will show different Imposter texture (Via pre-baked texture array's during the *VRSI\_CREATOR\_MULTIVIEW* process) depending on the angle of where they are being viewed from.

**PC\_MOBILE:** Optimized shaders and baking process for Using VRSI with PC and Mobile Projects.

- vrsiCamFacing\_CUTOUT\_DIRECTIONAL
- vrsiCamFacing\_CUTOUT\_DIRECTIONAL\_POINT
- vrsiCamFacing\_CUTOUT\_UNLIT
- vrsiCamFacing\_MSAA\_DIRECTIONAL
- vrsiCamFacing\_MSAA\_DIRECTIONAL\_POINT
- vrsiCamFacing\_MSAA\_UNLIT

*“Please note: The Default Shader attached to the prefab after generation for both static and camera facing is MSAA\_DIRECTIONAL, this can be changed to whatever your requirement by selecting any shaders in the VRSI shader list”*

Lighttable shaders will be Lit by all the lights in the scene, including casting Shadows.

All the shaders have the option for Wind Animation toggle, which can be used for most foliage.

**MSAA** must be enabled in your Projects Quality settings for all *Alpha-to-coverage* shaders to work. These shaders provide anti-aliasing to the Cutout at the texture level in VR and look much nicer.

All shaders include a Color variation toggle to provide minor color changes, which is suitable for Trees and other foliage where they can all look different.

*“**IMPORTANT:** Please disable Dynamic Batching in your project settings if you have enabled GPU Instancing in the VRSI Shader”*

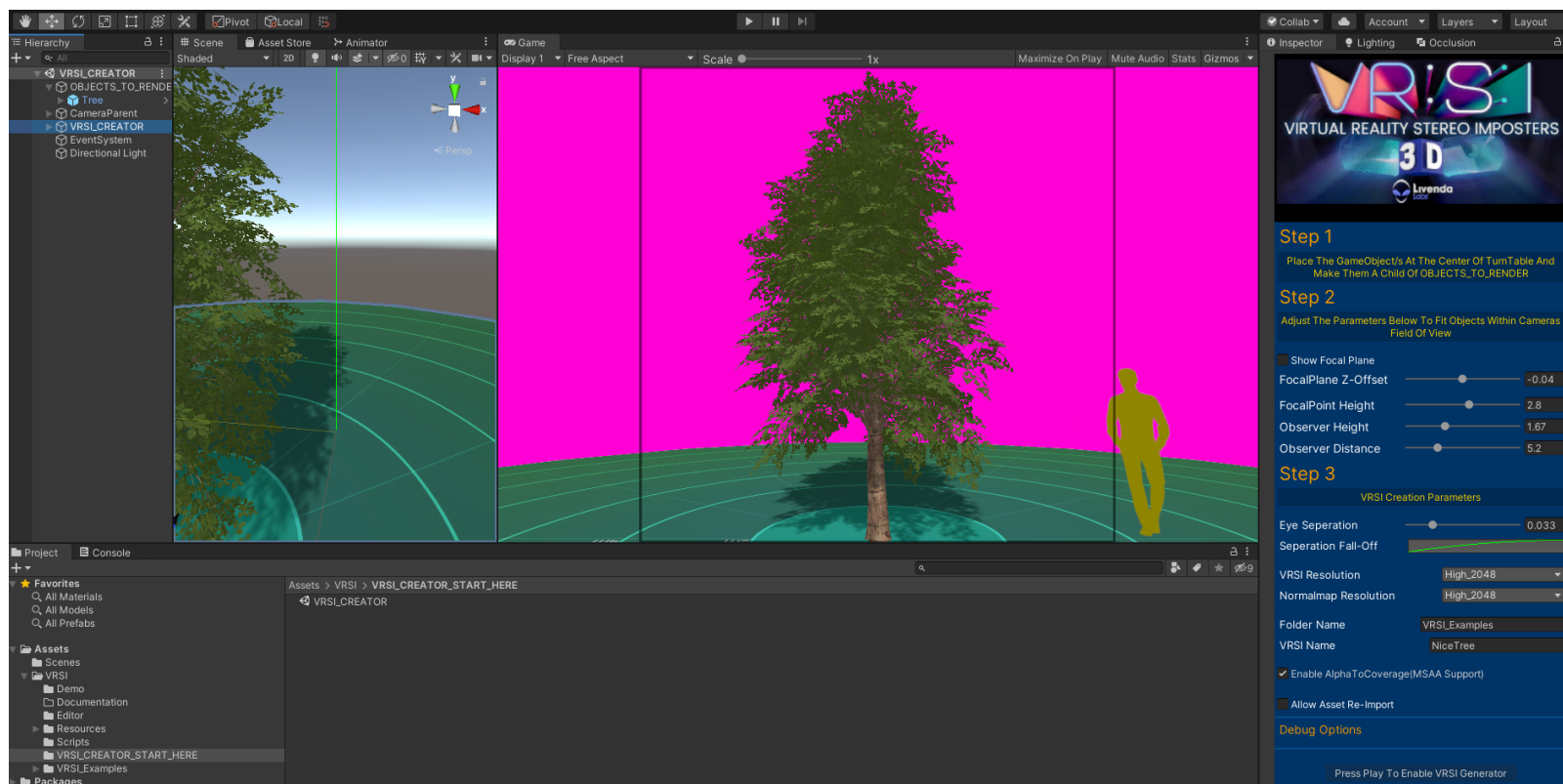
## **Quick Start Guide**

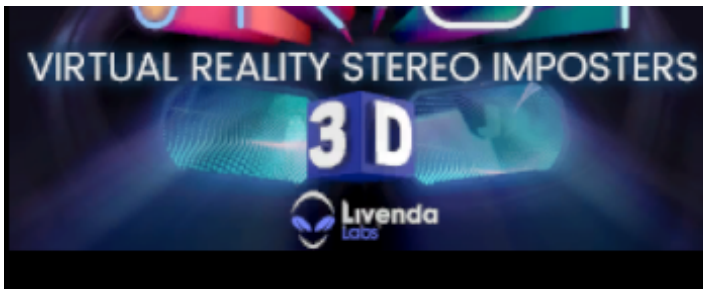
**IMPORTANT:** THE FIRST STEP IS TO OPEN THE IMPORT\_ASSIST FOLDER AND OPEN THE VRSI\_IMPORT\_ASSIST scene FILE and PRESS RUN. This process should be performed every time you install VRSI into a new VR PROJECT or if you move the generated VRSI's into a different project. This process ensures the generated VRSI textures maintain the correct IPD settings as Unity can sometimes corrupt textures during import.

A good methodology to work with VRSI is to create a fresh new project with only VRSI installed, and generate your required VRSI's and build up your library. When complete, just copy and paste the newly generated textures (the entire folder) into your own VR Project and run the VRSI\_IMPORT\_ASSIST scene by installing VRSI into your Project.

**VRSI** is extremely easy to use and as an editor tool, there are no other dependencies. Please note if you like to preview the generated VRSI's inside VR, you would need to set up your version of Unity for VR first. Please refer to the appropriate Unity guidelines on how to use your HMD with Unity.

After downloading the VRSI Package from the Unity Asset Store, just install VRSI. There are no project settings in VRSI, so it will not affect your current project.





## Step 1

Place The GameObject/s At The Center Of TurnTable And  
Make Them A Child Of OBJECTS\_TO\_RENDER

## Step 2

Adjust The Parameters Below To Fit Objects Within Cameras  
Field Of View

☐ Show Focal Plane

FocalPlane Z-Offset  -0.13

FocalPoint Height  2.4

Observer Height  1.71

Observer Distance  4.5

## Step 3

### VRSI Creation Parameters

Eye Separation  0.033

Separation Fall-Off

VRSI Resolution

Normalmap Resolution

Folder Name

VRSI Name

☒ Enable AlphaToCoverage(MSAA Support)

### Select The Versions To Generate

- ☒ Create For VR (Standard VRSI)
- ☐ Create For PC-MOBILE (Hight-Quality Billboards)

☐ Allow Asset Re-Import

Press Play To Enable VRSI Generator



After the installation is complete, just select the 'VRSI CREATOR START HERE' folder and open the 'VRSI CREATOR' Scene.

VRSI Creation is a simple three step process, the default scene already includes a Tree at the origin. This object can be replaced by any object you desire regardless of complexity! It will be baked by the VRSI process.

### The baking process and lighting your object?

VRSI will bake the selected object as is, what you see is what you get, this includes ambient light information, lights, light color, reflections and so on.

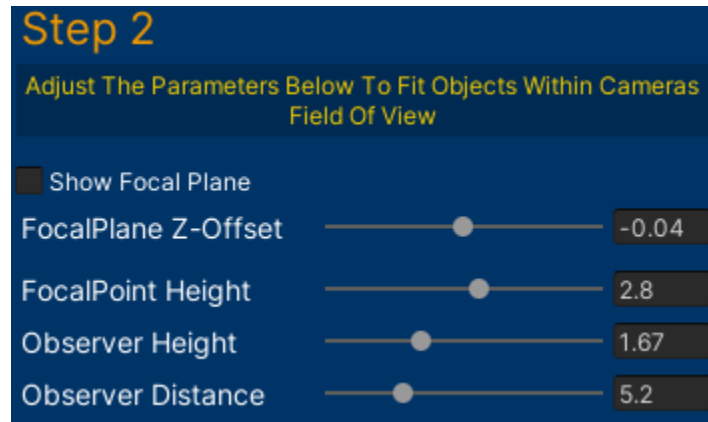
The VRSI Creator scene is set up to provide a '*neutral lighting environment*', the ambient environment light is Color only at mid grey and shadows from the directional light are at 50% strength. These are good settings for most objects as they provide detail and can be re-lit dynamically while looking almost identical to the reference 3D mesh.

You are Free to change your environment settings to whatever you like, including adding more lights, change Environment Lighting source, Environment Reflection source or it's strength to achieve your desired look.

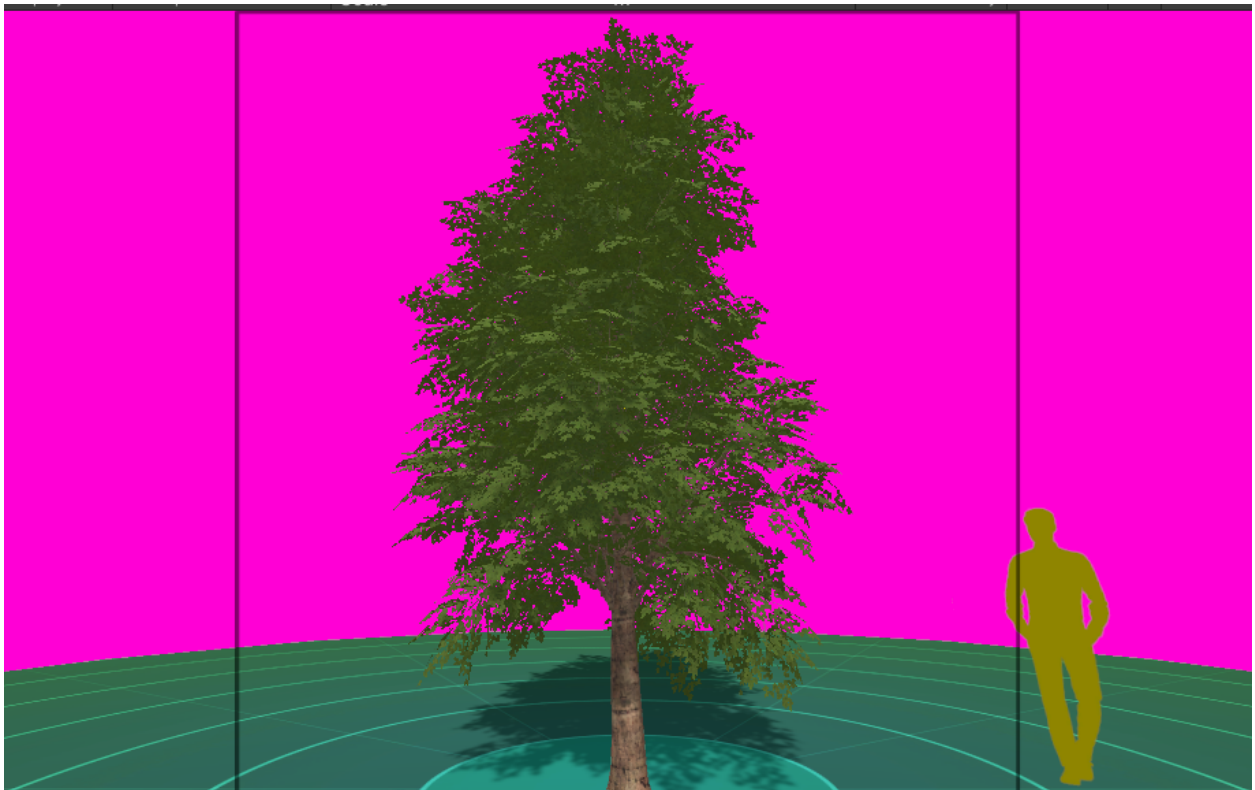
## Step 1

Import the object you like to convert to VRSI format and place it at the centre of the green-turntable, at world location 0,0,0 . Once it is centred, drag and drop the object onto the *OBJECTS\_TO\_RENDER* game objects from the scene Hierarchy , making it the child of *OBJECTS\_TO\_RENDER*. VRSI can bake any transparent material regardless of the shader it is using, it can be any shader.

## Step 2



In this step we primarily adjust values to fit the 3D mesh as tight and as close as possible to the camera while fitting 'inside' the rectangular bounds in the game viewport. As well as specifying appropriate general height and view direction specific to this object. Adjust the above parameters to fit the 3D mesh object as close and tight as possible within the square gameview square boundaries.



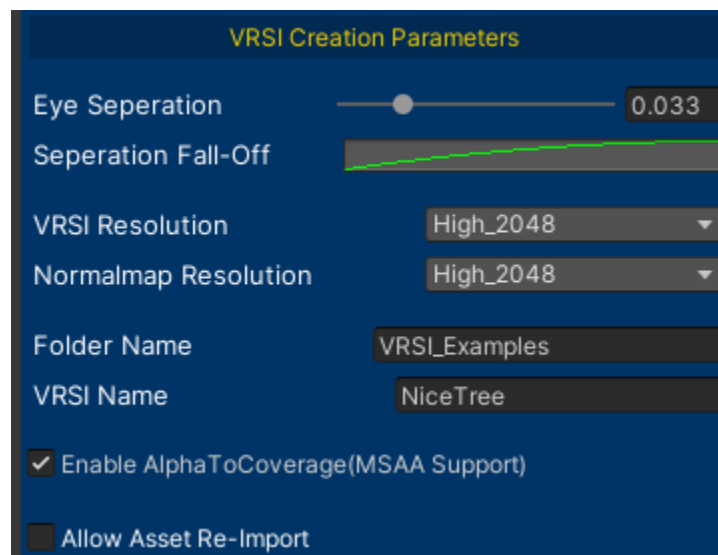
**Show Focal Plane:** Toggle ON/OFF the focal Plane

**Focal Plane Z-offset:** This value determines how much the object will lie in-front or behind the focal plane and where the object will be touching the ground. It is important to set this value as close as possible to the front where the object meets the ground. For example for a Tree type object, it should be just at the front-tip of the trunk. A value of 0 is usually good default for most objects. If an object has extremely protruding sections towards the camera, this can be adjusted accordingly.

**Focal Point Height:** This value will adjust how high or low the camera is looking at. Adjust this value as close as possible to where generally this object will be seen from most of the time.

**Observer Height:** The general height of the observer in the scene/game/app 1.6-1.8 m is a good average height

**Observer Distance:** The distance to the object from the camera, adjust this value to fit the object inside the rectangular bounds as tight as possible.



## Step 3

**Eye Separation:** The average eye distance (average IPD, interpupillary distance) for stereo parallax. A value of 0.033 is 6.6 cm, which is generally a good default value.

**Separation fall-off graph:** This graph will set how quickly the stereo pair eye separation/ parallax (interpupillary distance) strength will change according to how far the object is to the player camera.

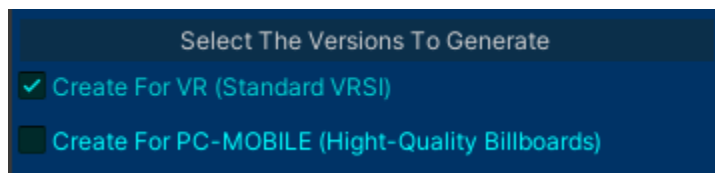
**VRSI resolution:** The horizontal resolution of the Albedo/Diffuse texture

**Normal-Map resolution:** The horizontal resolution of the normal map generated, used by relightable shaders.

**Folder Name:** The name of the folder inside your project where the generated VRSI will be placed. If the folder does not exist, a new folder will be created.

**VRSI Name:** The name of the generated VRSI

**Enable Alpha-to-Coverage:** When toggled on, the generated VRSI prefab will have the ability for Alpha-to-coverage. Please note MSAA must be enabled in your project quality settings for Alpha-to-coverage to work. Alpha to coverage is a multisampling computer graphics technique that replaces alpha blending with a coverage mask. This achieves order-independent transparency for when anti-aliasing or semi-transparent textures are used.



**Select The Version To Generate:** Select if you like to use VRSI for VR or Non-VR such as PC & Mobile or Both. If you generate for PC & Mobile, VRSI will

automatically generate an additional folder named PC\_Mobile where the generated VRSI prefabs can be found and used immediately.

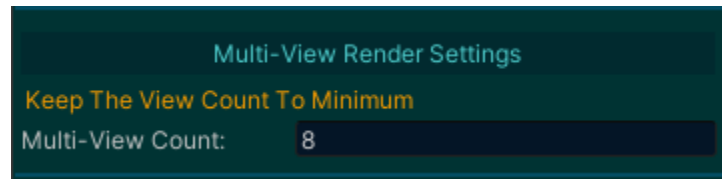
**Allow asset re-import: important**, When you are moving the generated VRSI to a different project to reuse, sometimes Unity will override the texture and re-calculate mip-maps, this will destroy the VRSI stereo pair distance information baked into the texture. If this is the case, just load the VRSI creation studio and enable this value. After doing so you can just go to the individual VRSI and “re-import” both the Albedo texture (normally the name of the VRSI) and it’s corresponding normal map, denoted by \_norm. **You do not need to import any of the .mip textures.**

## You're Now Ready to Create a VRSI!



After completing the steps above, just press play in Unity and you will see the above buttons. Just Click CREATE VRSI and in a few seconds the VRSI will be generated and placed in the folder designated. Once it is generated, if you have your VR device setup and ready in Unity, you can do a quick preview by pressing the START VR PREVIEW button.

## Multi-View VRSI Creation



As of V1.2 VRSI allows you to generate a true stereoscopic imposter that shows the correct and appropriate imposter texture depending on where they are being viewed from! This is a separate Creation process than the standard method, it is however essentially the same procedure. You now have an option to specify the number of Views around 360° to render, into a single Texture Array 3D Stereoscopic VRSI. Please note: These can be quite large in memory consumption.



A VR Demo of Multi-View can be found in the Demo Folder of VRSI called *MultiView\_VRSI\_Demo\_Experimental*.

## FAQ

### Which render path is VRSI Compatible with?

V1 is for Standard render path. HDRP and URP compatibility will be available very soon. However as VRSI is a Unity Tool, compatibility with HDRP and URP is easy to implement as only the generated shaders require changing, the tool can be already used inside URP or HDRP if you like to write your own shader until we release complete compatibility with subsequent updates.

### Is GPU instancing supported?

Yes, all shaders have the option to enable GPU Instancing, please note, you need to disable Dynamic Batching in your project settings if you have enabled GPU Instancing in the VRSI Shader.

### Do you have any recommendations?

For Oculus Quest Native apps we strongly recommend Vulkan render path and Linear lighting.

## Support Contact

Please email for any further help or enquiry [support@livenda.com](mailto:support@livenda.com)

For more information and updates or upcoming products please visit [www.livenda.com](http://www.livenda.com)

# VRSI: VIRTUAL REALITY STEREO IMPOSTERS

Copyright © 2021 Livenda Labs Pty Ltd

## *Change Log*

=====

### **VRSI 1.2**

=====

1) Added Support to VRSI Creation Studio To Render High-Quality PC & MOBILE Ready Billboards. You can now toggle this option as a new Variant. There are also numerous new shader choices available for PC & MOBILE variants. More PC & MOBILE features and shaders are planned for the next release.

2) We are introducing the all new VRSI\_CREATOR\_MULTIVIEW option in this release (Located In VRSI\_CREATOR\_START\_HERE folder). VRSI\_CREATOR\_MULTIVIEW simplifies the process of creating VRSI's that have a distinct number of views (Multi-View).

Multi-View Creation Process is essentially the same as a regular VRSI with the added option of view Count. This is the number of distinct views in 360



Degrees around the object. There are also accompanying shader options available in the Resources library.

SetGlobalCamPos.cs is a new script required by Multi-View Renderer.

TeddyBear Asset is now included in the resources folder, also used in VRSI\_CREATOR\_MULTIVIEW

3) All VRSI shaders have been updated to correctly support Single-Pass Stereo Rendering Mode in all VR modes including SteamVR.

4) Various Code Fixes and new supporting assets included

5) NEW Specular shader added to STATIC VRSI's, great for assets requiring Specular highlight such as props and rocks (complete PBR solution is coming soon)

6) Documentation update for Multi-View Creation process and PC & Mobile Use.

=====

## **VRSI 1.1**

=====

1) Added two new demos

**\*\* MultiView\_VRSI\_Demo\_Experimental \*\***

This is a basic demo of Upcoming feature called Multi-View VRSI or MV-VRSI. The demo shows a Teddy bear with 16 VRSI packed into a single Texture Array. This allows

The VRSI has multiple viewpoints, you can essentially walk around it. We will include a new Tool in the next release of VRSI to make it simple to create Multi-View VRSI's.

**\*\* VRSI\_LOD\_Demo \*\***

This demo shows how you can easily replace the traditional billboards in LOD groups with VRSI alternatives!. There are over 7000 Trees in the scene, that are GPU instances.

2) Clean-up of VRSI Shaders and new Specular VRSI shaders added.

3) Two new VRSI's have been added to the VRSI\_Examples

*Teddy*

*PineTree* (The original 3D version has over 700K triangles)

4) Added vrsiCamFacing\_MSAA\_DIRECTIONAL\_BARE shader

This shader had no Wind or Color Variation support, so it's bare minimum. More Variants of Bare shaders will be released in the next update

5) Import Assist Has Been Updated

