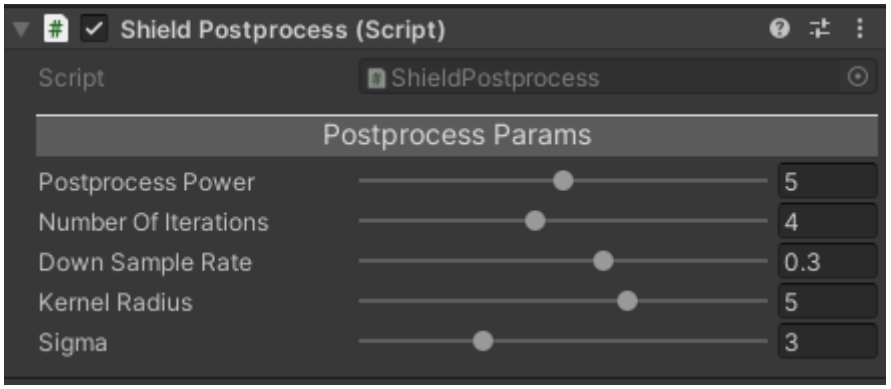
Found a bug, need a feature or help with implementation. Contact at:
fx.valley.contact@gmail.com

Installation & Setup
v2.0

1. **BuiltIn Render Pipeline**

- Asset is ready to use after importing.

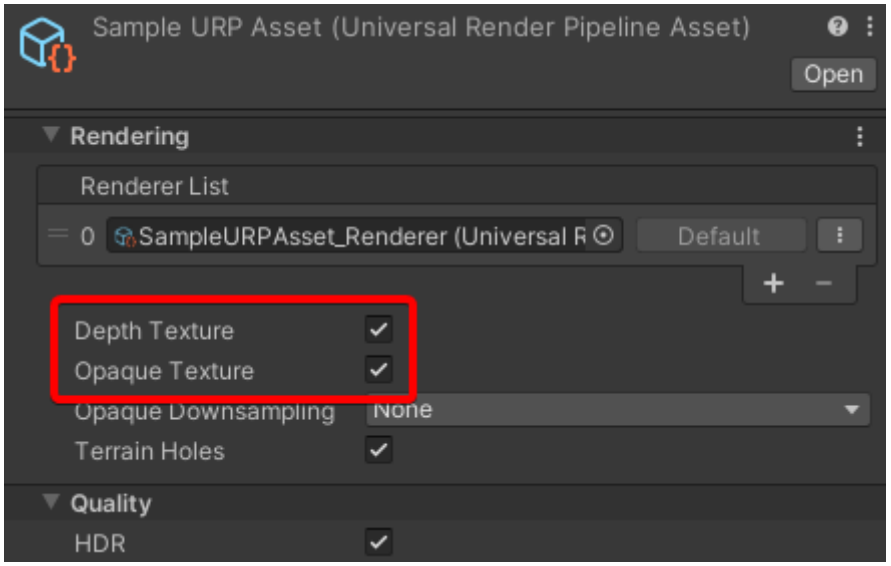
- To make everything work on your scene add *FXV/ShieldEffect/Scripts/ShieldPostprocess.cs* script to the camera:



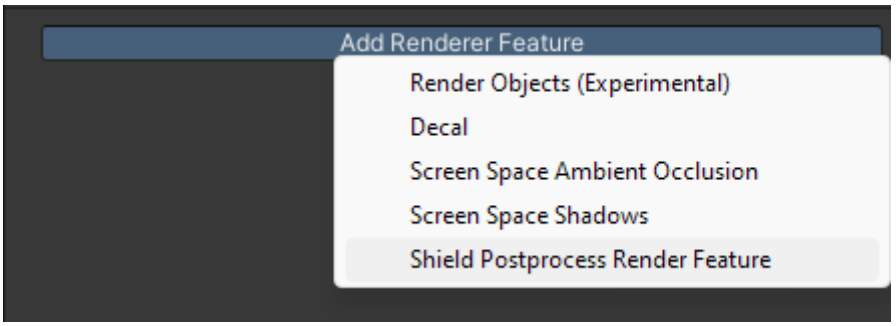
2. **Universal Render Pipeline**

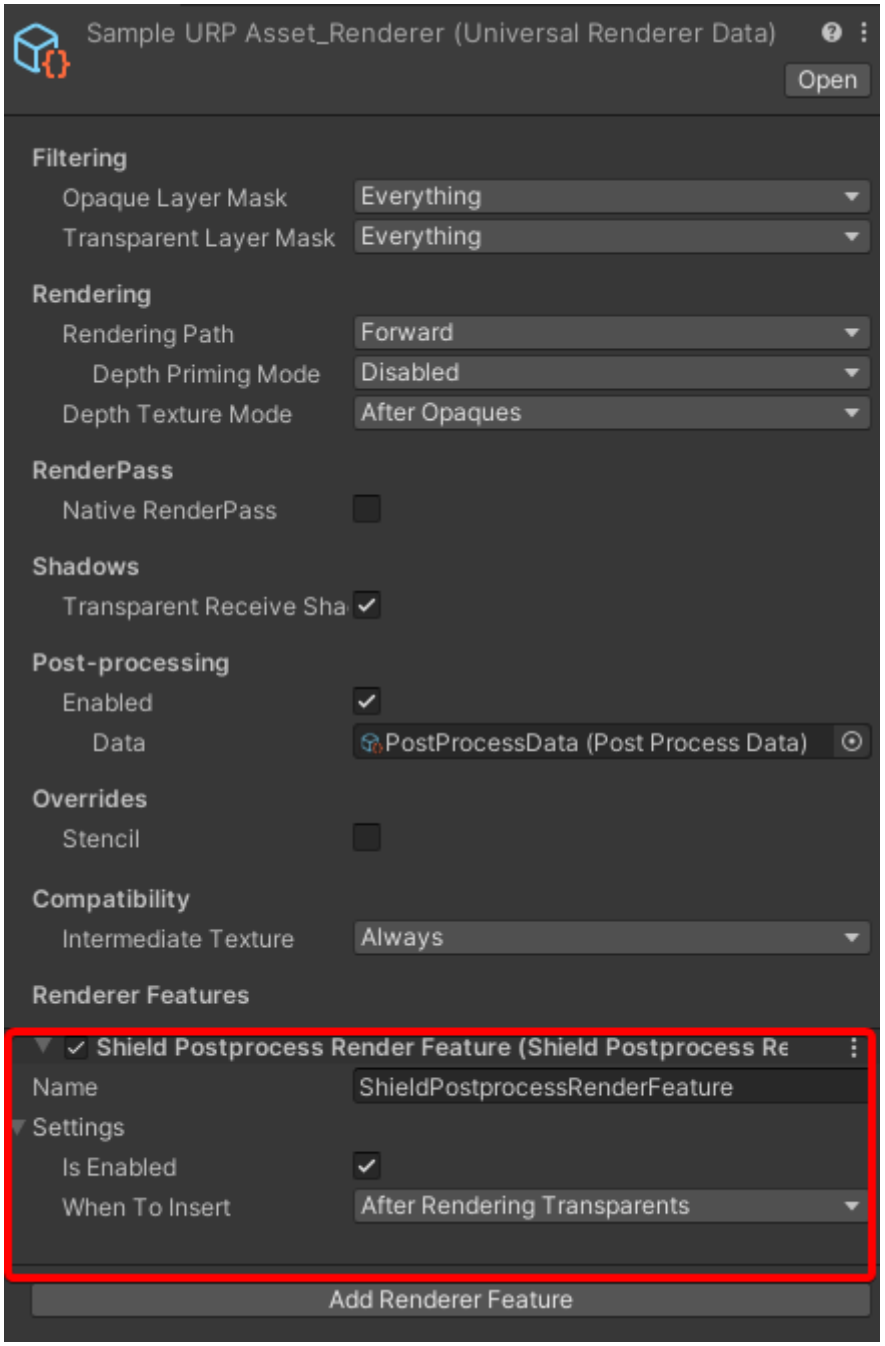
- After importing you will need to install the URP package *FXV/ShieldEffect/InstallURP.unitypackage* that contains all required assets.

- For all features to work properly depth texture and opaque texture needs to be enabled in *Universal Render Pipeline Asset:*

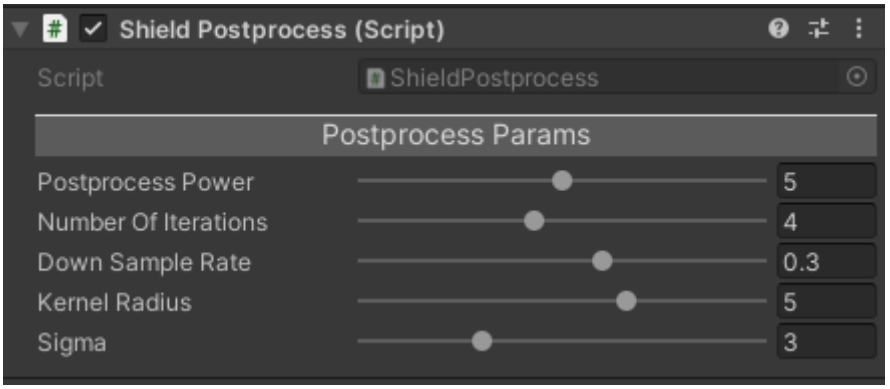


- To enable postprocess add *ShieldPostprocessRenderFeature* to Universal Render Pipeline Renderer. Set it to render After Transparents.





- To make everything work on your scene add *FXV/ShieldEffect/Scripts/ShieldPostprocess.cs* script to the camera:



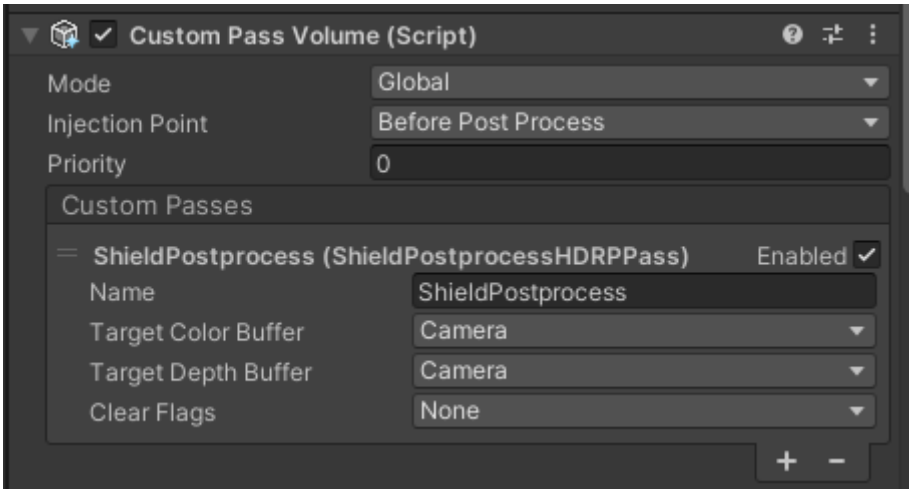
-

3. High Definition Render Pipeline

- After importing you will need to install the HDRP package *FXV/ShieldEffect/InstallHDRP.unitypackage* that contains all required assets.

- To make everything work on your scene add *FXV/ShieldEffect/Scripts/ShieldPostprocess.cs* script to the camera.

- Also add *Custom Pass Volume* with *Shield Postprocess HDRP Pass* configured as on screenshot below*:*



Package contents

- Each render pipeline has its own demo scene:

  BuiltIn:
  FXV/ShieldEffect/Demo/Demo.unity

URP:
FXV/ShieldEffect/URP/Demo_URP.unity

HDRP:
FXV/ShieldEffect/HDRP/Demo_HDRP.unity

- There are multiple prefabs and materials that you can use right away or use as a base for tweaking own variations:

  FXV/ShieldEffect/Prefabs

- All the prefabs and materials are presented on the demo scene. When in play mode you can change type using a leaver in the center and turn on/off each shield using a leaver near the shield object.

- For hit effect you can shoot using the left mouse button. Hold left ctrl for a bigger bullet to test different hit effect sizes.

- There is special scene with setup for custom rim texture baking when using custom meshes:

  FXV/ShieldEffect/Demo/CustomTextureBaking.unity

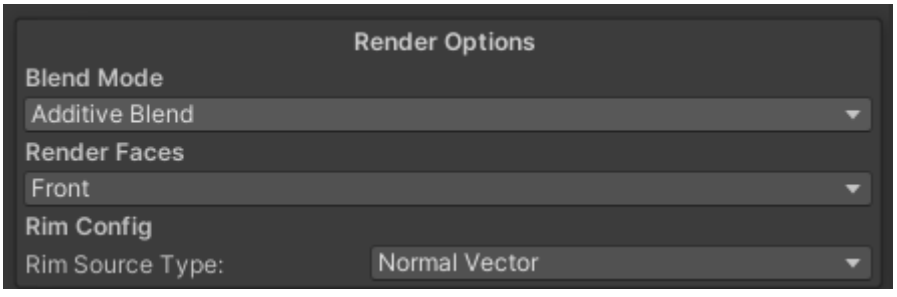
Using Shield Effect

1. **Setting up new object**

- Create material and assign to it FXV/FXVShieldEffect shader
- Create new object with mesh renderer and set the newly created material in its Renderer
- Add Shield.cs script to the object

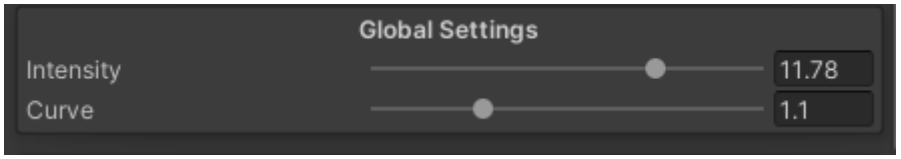
2. **Tweaking material properties**

- **Rendering options**



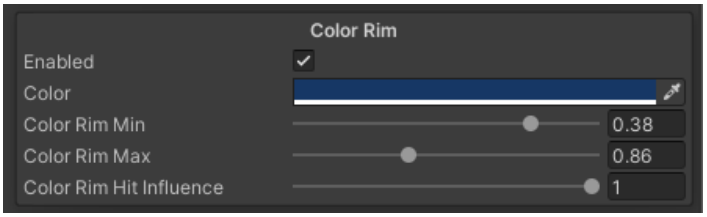
Blend Mode	select blending mode alpha or additive
Render Faces	option to render back faces - in situations if rendering inside of the shield is required
Rim Config	what is the source for rim calculation: - **Normal** - rim will be aligned to view space normal - **Texture** - rim will be calculated from provided texture (R - rim color, G - texture rim, B- pattern rim)

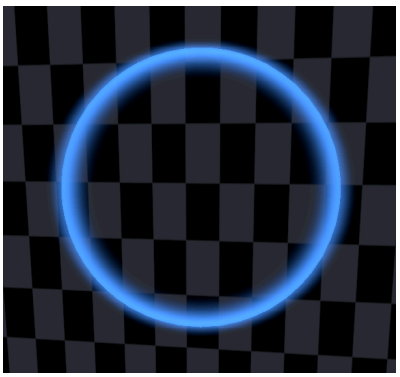
- **Global Settings - settings for final tweaking of effect intensity**



Intensity	final multiplier of output color
Curve	shape of the fade curve

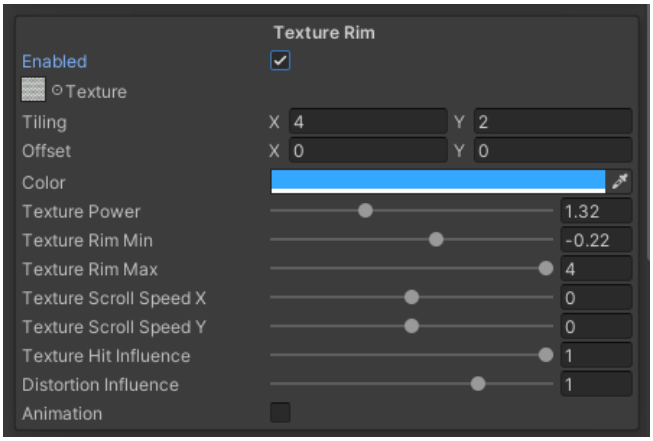
- **Color Rim - adds solid color rim effect**

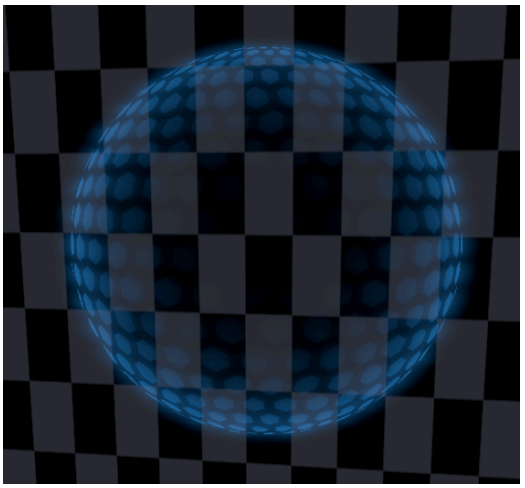


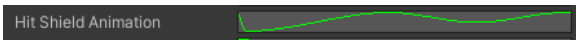


Color Rim Min/Max	size of the rim effect
Color Rim Hit Influence	how much the Hit Shield Animation curve from Shield script component affects it.

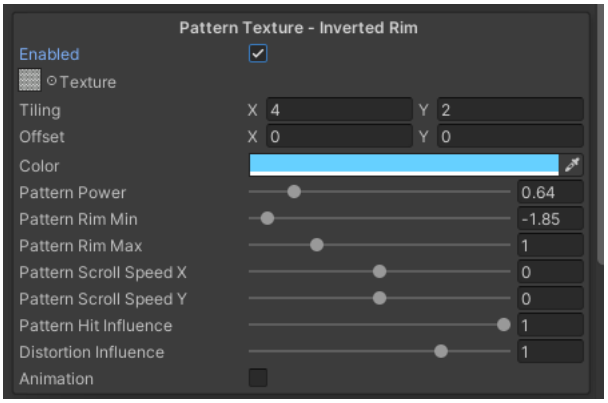
- **Texture Rim - add rim effect with texture**

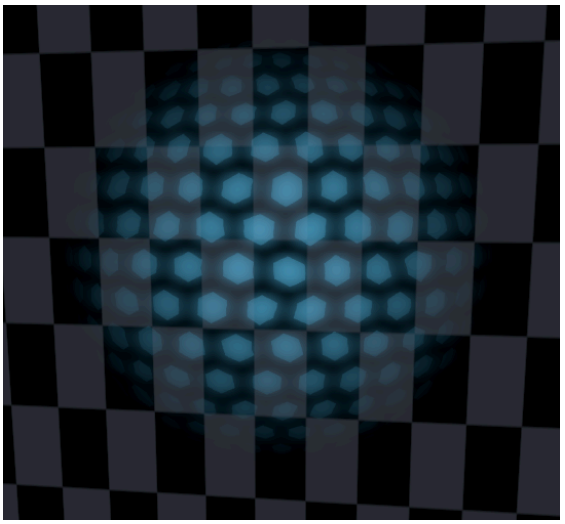


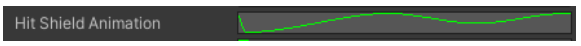


Texture Rim Min/Max	size of rim effect
Texture Power	how much to multiply final color
Texture Scroll Speed X/Y	speed of texture position scroll over time for X/Y axis
Texture Hit Influence	how much the Hit Shield Animation curve from Shield script component affects it 
Animation	Sin fade animation based on texture value
Distortion Influence	(available when Distortion is on) how much distortion affects this texture

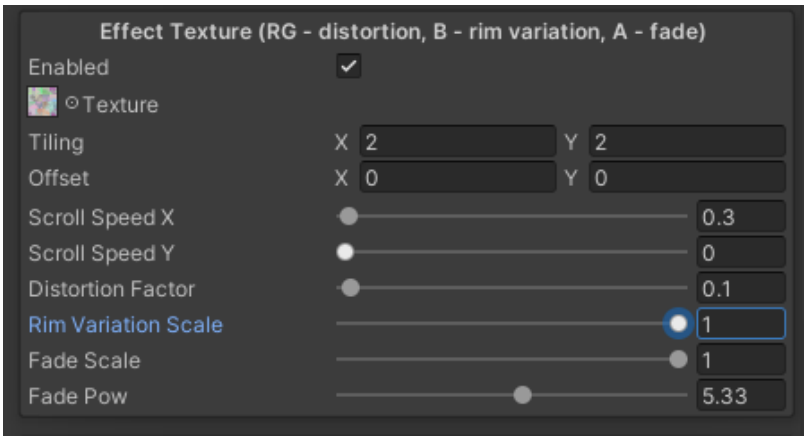
- **Pattern Texture - add inverted rim effect with texture so it will add nicely with Texture and Color Rim**

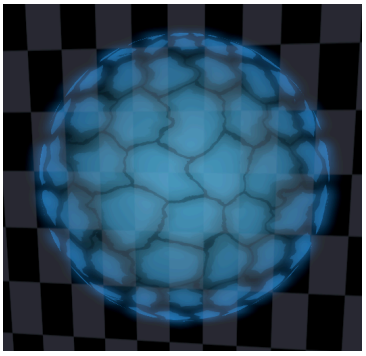
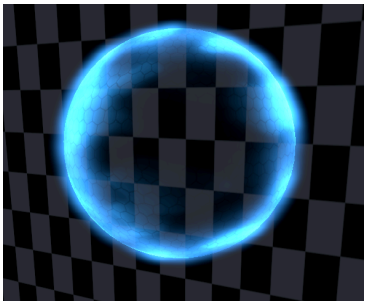


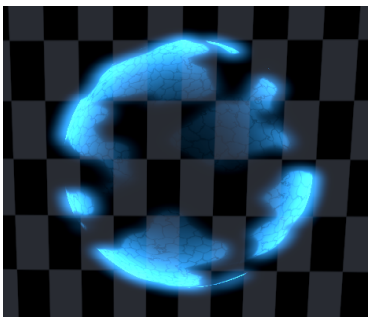


Pattern Rim Min/Max	size of rim effect (inverted)
Pattern Power	how much to multiply final color
Pattern Scroll Speed X/Y	speed of texture position scroll over time for X/Y axis
Pattern Hit Influence	how much the Hit Shield Animation curve from Shield script component affects it 
Animation	Sin fade animation based on texture value
Distortion Influence	(available when Distortion is on) how much distortion affects this texture

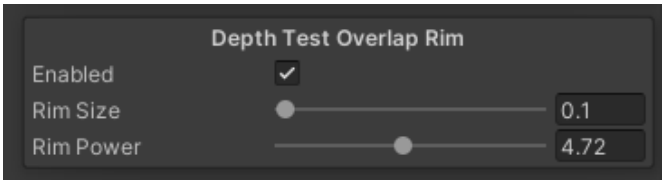
- **Effect texture - special texture with Red and Green chanell for distortion, Blue channel for rim effect variation, Alpha channel for fade out effect**



Speed X/Y	how fast texture will move over time
Distortion Factor	how much to distort by red and green channel of this texture (globally) 0 - off 
Rim Variation Scale	how much rim effect will be scaled by blue channel of this texture, 0 - off 
Fade Scale	how much to fade the shield effect with alpha channel of this texture, 0 - off

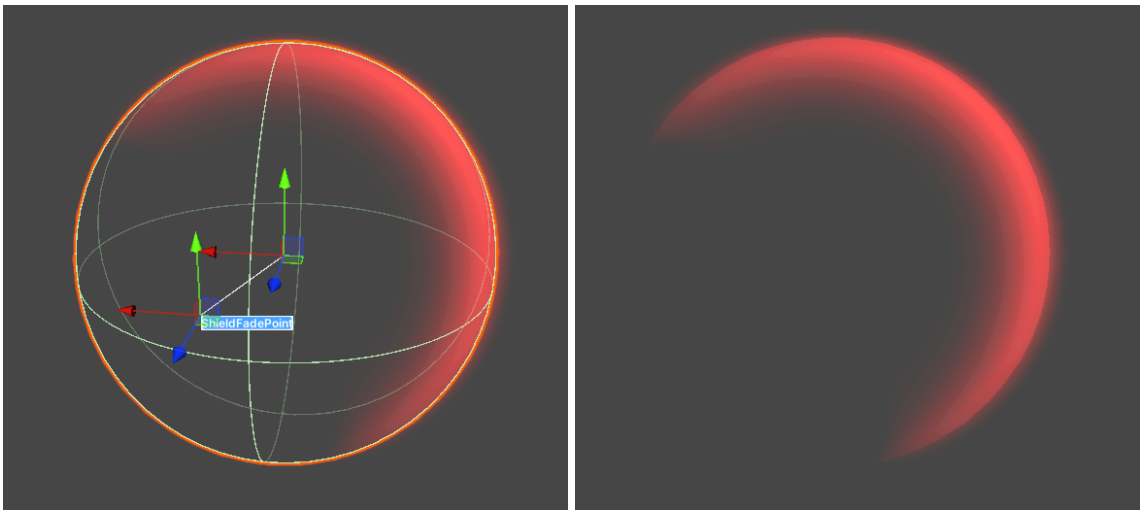
	
Fade Pow	change the edge/curve of the fade effect

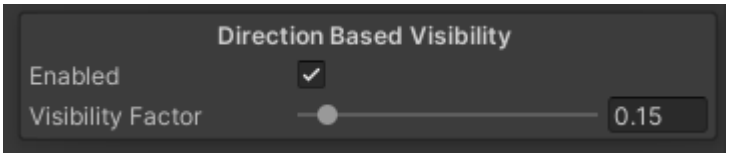
- **Depth Test Overlap Rim - enable color rim for intersections with geometry - this one needs to have Color Rim enabled**

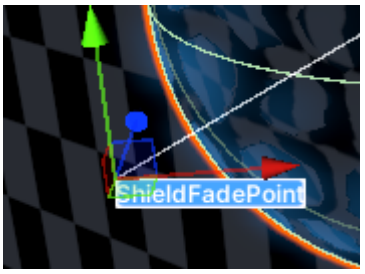


Rim Size	set size of the effect based on depth
Rim Power	how much to multiply color from Color Rim

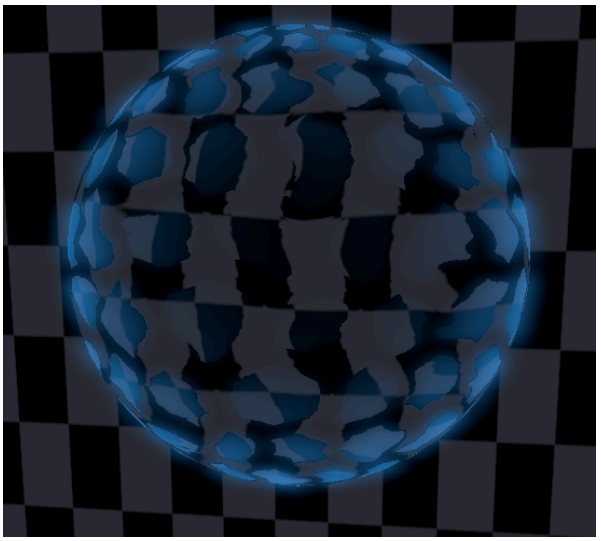
- **Direction Based Visibility - adds nice fade effect based on ShieldFadePoint position**





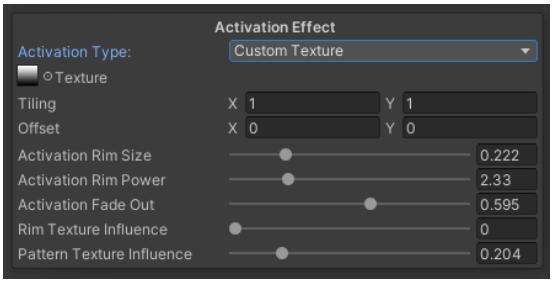
ShieldFadePoint (scene view)	position on scene - for fade direction - this point is visible in scene view when shield object is selected 
Visibility Factor	parameter for fade size

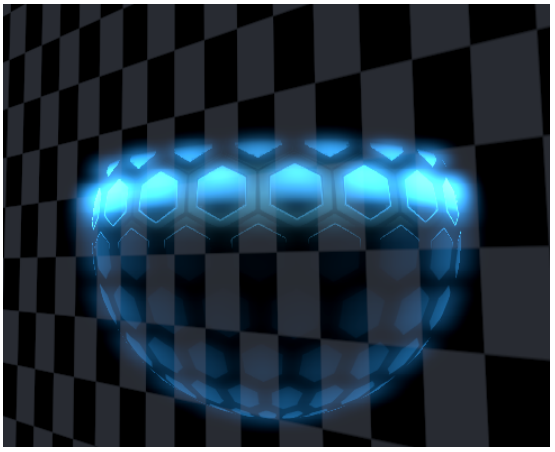
- **Refraction - distorts everything behind the shield**



Refraction scale	how much to distort
Refraction Rim Min/Max	scales refraction strength based on rim effect parameters
Refraction Background Exposure	lightens or darkens the background
Refraction Color/Texture/Pattern Influence	specify how much each layer affects the distortion effect

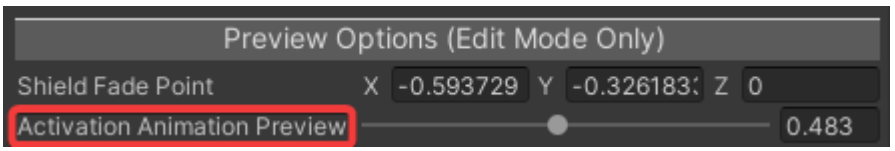
- **Activation Effect - animation effect when shield is turned on/off**



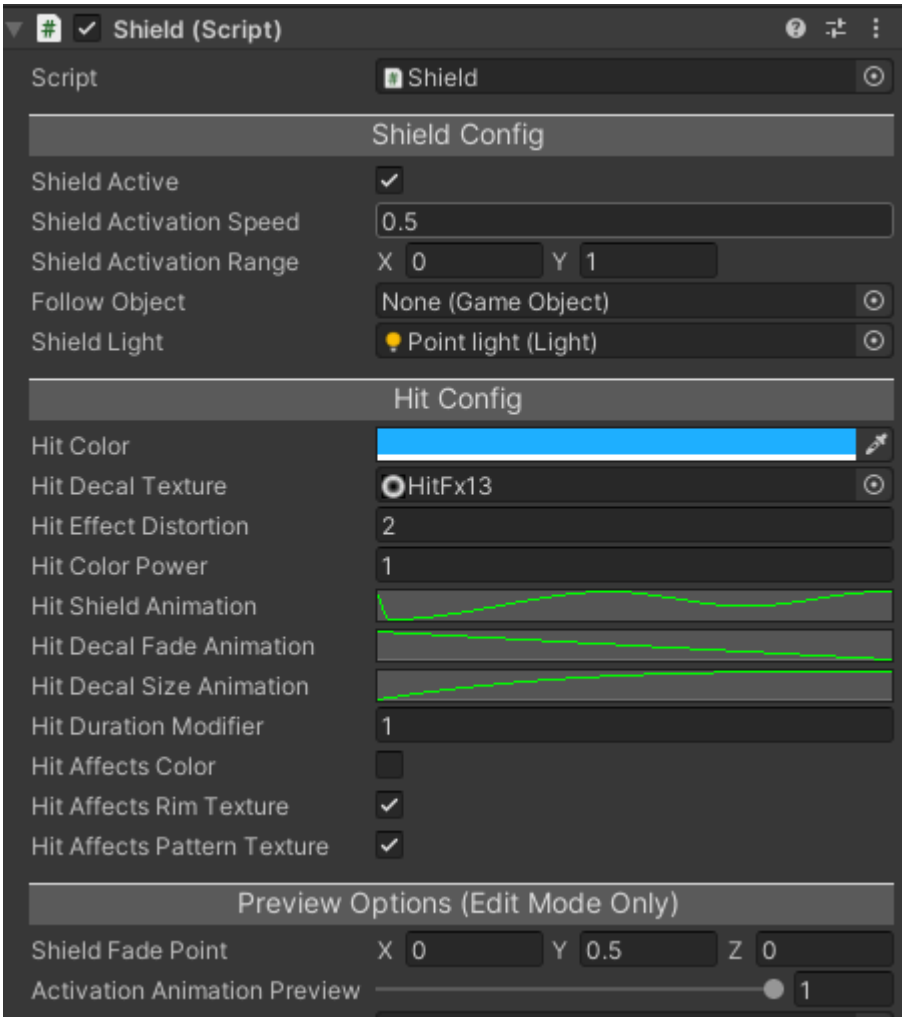


Activation Type	Select the source for activation animation - this will be compared to activation time for rim effect calculation: - **FinalColor** - final material color - **FinalColor_and_UVx** - final material color mixed with UV.x value - **FinalColor_and_UVy** - final material color mixed with UV.y value - **CustomTexture** - custom texture mapped onto object - **UVx** - uv.x value - **UVy** - uv.y value
Activation Rim Size	size of the rim effect
Activation Rim Power	how much rim color will be multiplied
Activation Fade Out	when (in time) to start fade out of rim effect 0 - no fadeout
Rim Texture Influence	how much rim texture will have effect on activation time
Pattern Texture Influence	how much pattern texture will have effect on activation time

For preview and tweaking purposes there is slider for activation animation preview in Shield component:



3. Configuring Shield Component

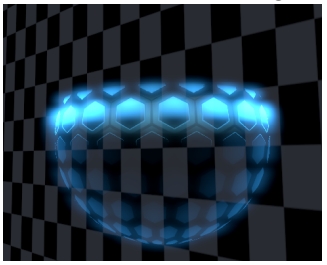


- Shield Config

Shield Active	is shield active when the scene is started in play mode
Shield Activation Speed	how fast the activation animation will be
Shield Activation Range	specify activation min max values so that the shield is fully invisible at time 0, and fully visible at time 1. Use the Activation Animation Preview slider to tweak this.
Follow Object	in situation when shield cannot be a child of an object, specify transform here that the shield should follow (can be empty)
Shield Light	light that the shield should affect when turning on/off (can be empty)

- Hit Config

Hit Color	color of hit decal
Hit Decal Texture	texture for decal pattern, if empty hit will have attenuation calculated by it's radius
Hit Effect Distortion	how much hit will distort the shield
Hit Color Power	intensity of hit decal final color
Hit Shield Animation	curve for global shield color influence when shield is hit. this might be scaled for each layer individually by using **Color Rim/ Texture/Pattern Hit Influence** sliders in material
Hit Decal Fade Animation	the curve for fade animation over hit effect life time
Hit Decal Size Animation	the curve for size animation over hit effect life time. Affects **hitScale** from OnHit(Vector3 hitPos, float hitScale, float hitDuration)
Hit Duration modifier	modifier for hit lifetime, use when for example. Affects **hitDuration** from OnHit(Vector3 hitPos, float hitScale, float hitDuration)
Hit Affects Color/Texture/Pattern	set if hit effect should be mixed together with each layer of shield effect material

Shield Fade Point	manually set ShieldFadePoint position 
Activation Animation Preview	use this to visualize and tweak activation effect on scene during edit mode 

4. **Scripting**

class: Shield.cs

Core class of shield effect instance, handles enabling disabling and hit effects.

public void SetMaterial(Material newMat)
- Set new material at runtime, material should use FXShield shader

public void SetRimColor(Color c)
- *Set rim color at runtime*

public void SetTextureRimColor(Color c)
- *Set texture rim color at runtime*

public void SetPatternColor(Color c)
- *Set pattern texture color at runtime*

public void SetHitColor(Color c)
- *Set hit color at runtime*

public bool GetIsShieldActive()
- *Return true if shield is in active state*

public bool GetIsDuringActivationAnim()
- *Return true if activation animation is in progress*

public void SetShieldVisuallyActive(bool active)
- *Set shield effect rendering on/off but leaves collider intact*

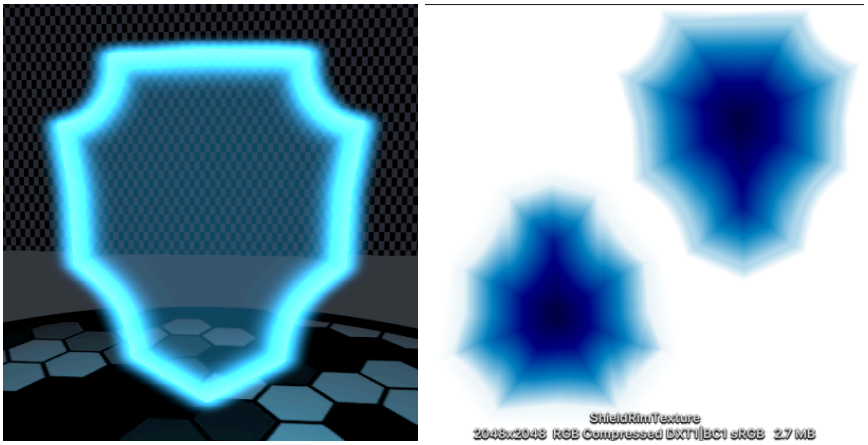
public void SetShieldActive(bool active, bool animated = true)
- *Turns shield on/off both visually and physically. Use animated = true fr activation animation, or animated = false for instant state change without animation*

public void SetShieldEffectFadePointPosition(Vector3 localPos)
- *Sets ShieldFadePoint position at runtime*

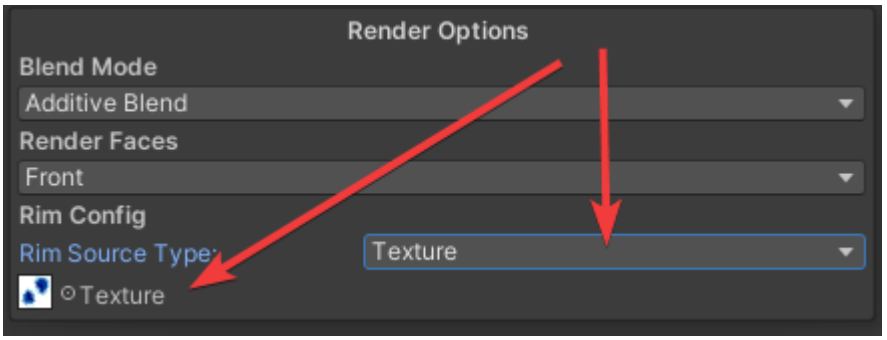
public void OnHit(Vector3 hitPos, float hitScale, float hitDuration)
- *Spawn hit effect at position, with specified scale and duration. Parameters will be multiplied by curves defined in Shield component hit config.*

5. **Baking Custom Rim Texture**

When using custom meshes, calculating rim effect with normal might not give good results. Here comes the **ShieldRimTextureBake** component. It bakes distance from the edge to texture from visible given BakeOrigin point:



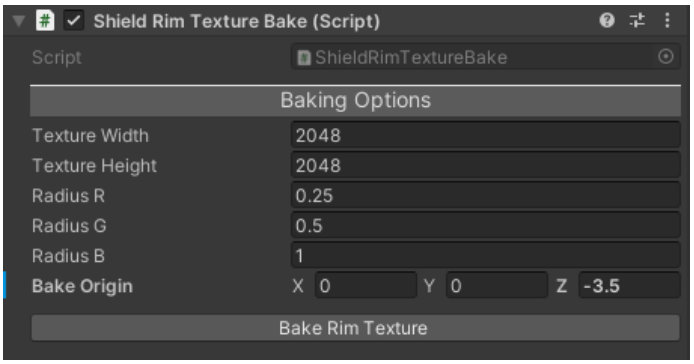
This texture should be used together with Texture - Rim Source Type:

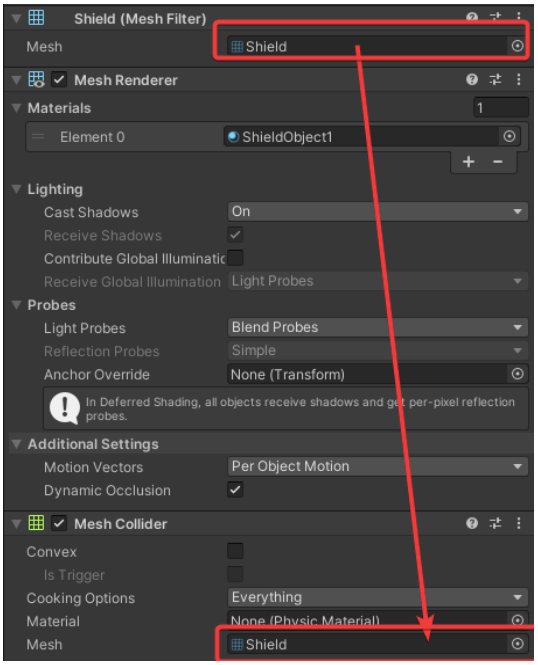


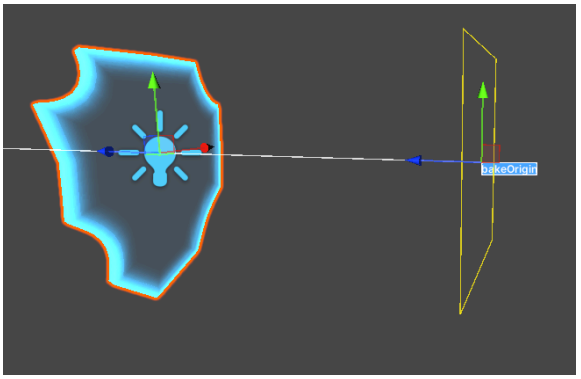
Each texture channel will be used for different rim value calculation:

- Red - color rim
- Green - texture rim
- Blue - pattern rim

To bake rim texture add ShieldRimTextureBake component to the object , and set the **mesh collider** with the same mesh as the rendered object:





Texture Width/Height	size of the texture (big sizes might take time to bake)
Radius R/G/B	rim size for each layer
Bake Origin	point from which the rim edge is viewed for baking (probably somewhere in front of mesh) 

Look at FXV/ShieldEffect/Demo/CustomTextureBaking.unity for sample setup.

It's good to do baking on an empty scene so that other colliders will not interfere with the process.