

Texture Material Expressions

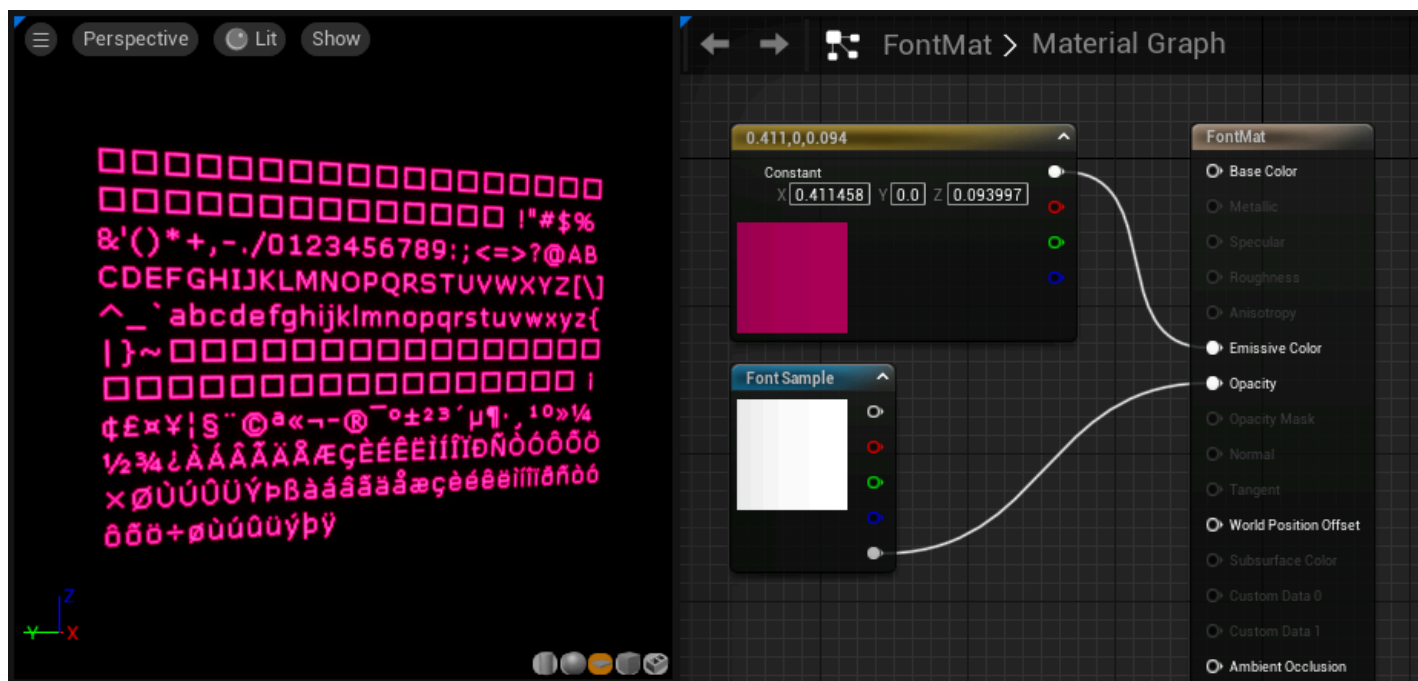
Expressions that sample and output textures.



FontSample

The **FontSample** expression enables you to sample the texture pages out of a font resource as regular 2D textures. The alpha channel of the font will contain the font outline value. Only valid font pages can be specified.

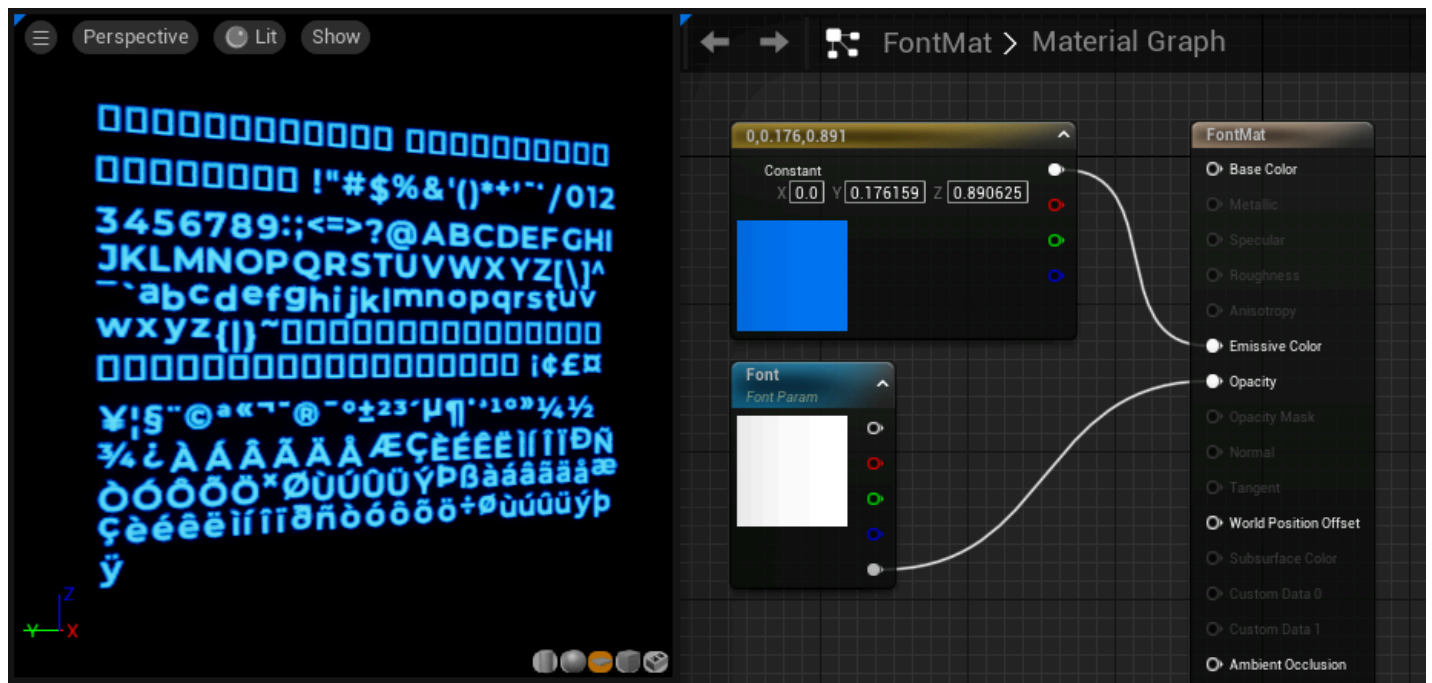
Item	Description
Properties	
Font	Holds the default font asset (from the Content Browser) to be held within the expression.
Font Texture Page	The current font texture page to be used as a part of the texture.



FontSampleParameter

The **FontSampleParameter** expression provides a way to expose a font-based parameter in a Material Instance Constant, making it easy to use different fonts in different instances. The alpha channel of the font will contain the font outline value. Only valid font pages can be specified.

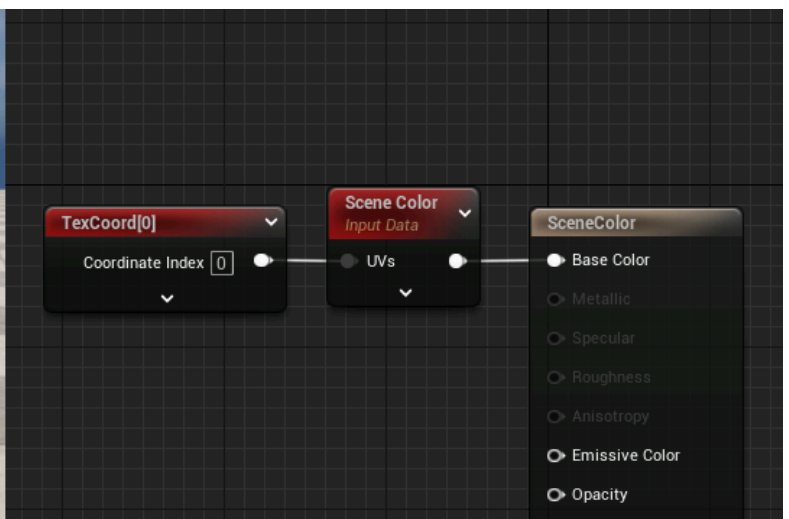
Item	Description
Properties	
Parameter Name	Specifies the name used to identify the parameter in instances of the Material and through code.
Group	Provides a way to organize parameter names into groups, or categories, within a MaterialInstanceConstant. All parameters within a Material that have the same Group property name will be listed underneath that category in the Instance Editor.
Font	Holds the default font asset (from the Content Browser) to be held within the expression.
Font Texture Page	The current font texture page to be used as a part of the texture.



SceneColor

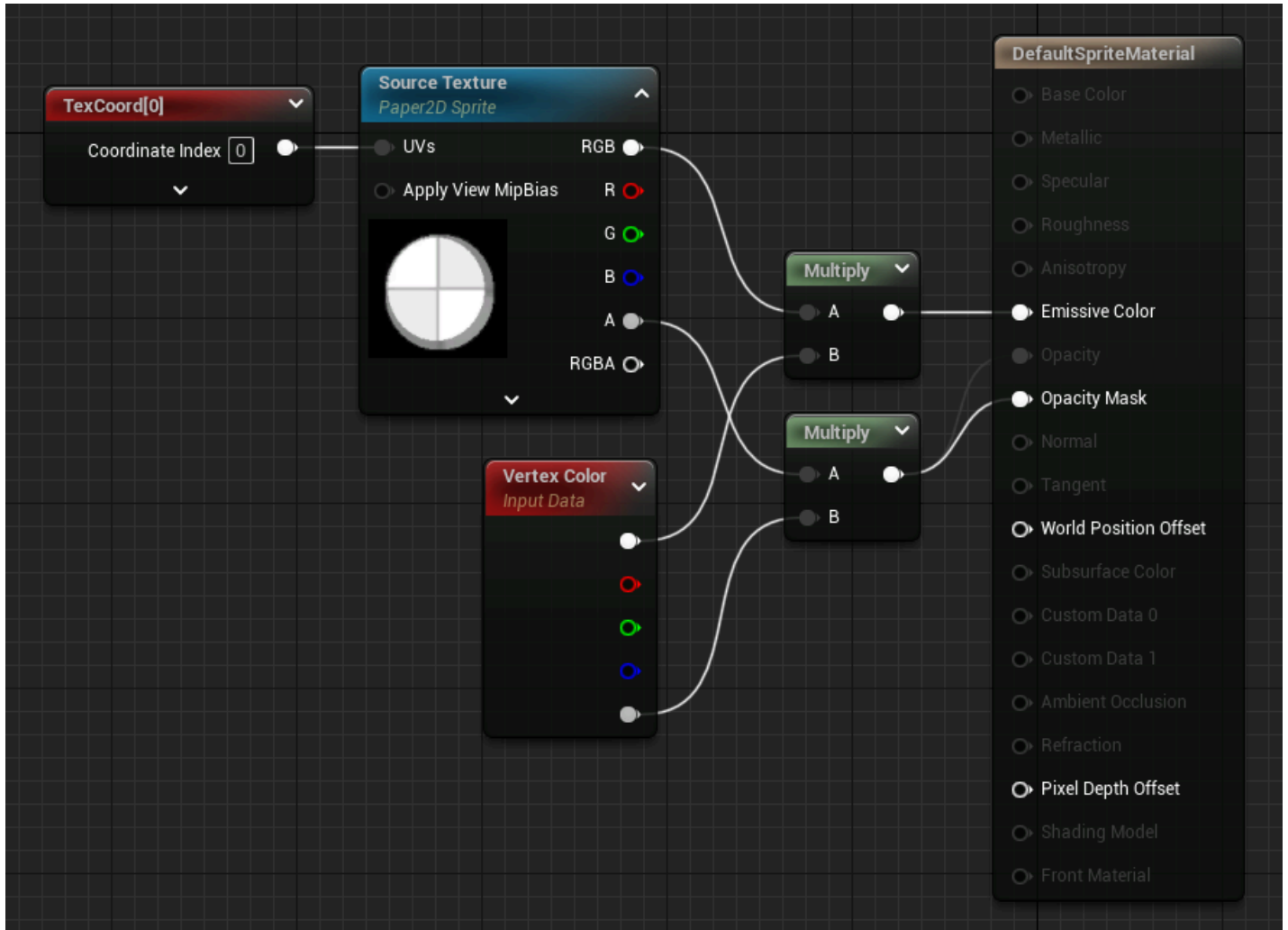
The **SceneColor** Material expression outputs the existing scene color.

Property	Description
OffsetFraction	Takes in a 2D vector that offsets the scene color in screen space. Setting this to either 0,1 or 1,0 will provide a slightly brighter version of the scene color. Setting it to 1,1 will provide an even brighter result.



SpriteTextureSampler

A custom sprite Material can be created by duplicating one of the existing ones, or creating a new Material in the **Content Browser**. When a sprite is rendered, the texture defined in a sprite asset will be piped into any texture parameters named "SpriteTexture" in the Material. The SpriteTextureSampler node can be placed to do this automatically.



Sprite instances will pass in their color as a vertex color, but it can be used for anything in the material, not just tinting.

Property	Description
Texture	Specifies the texture sampled by the expression. To set the texture, first select a texture in the Content Browser . Then, select the Texture property in the expression's property window and click the 'Use Current Selection' button.

Property	Description
Sampler Type	The type of data that will be sampled and output from the node.
MipValueMode	Selects how to customize the sample's mip-level or derivatives from the default hardware computed. Affects the look and performance.
Inputs	
UVs	Takes in UV texture coordinates to use for the texture. If no values are input to the UVs, the texture coordinates of the mesh the material is applied to are used.
Outputs	
RGB	Outputs the three-channel RGB vector value of the color.
R	Outputs the red channel of the color.
G	Outputs the green channel of the color.
B	Outputs the blue channel of the color.
A	Outputs the alpha channel of the color. If a texture does not contain an alpha channel, connecting the 'alpha' channel to something, while not technically illegal, will always result in zero (black).

TextureObject

The **TextureObject** expression is used to provide a default texture for a texture function input within a function. This node does not actually sample the texture, so it must be used in conjunction with a TextureSample node.

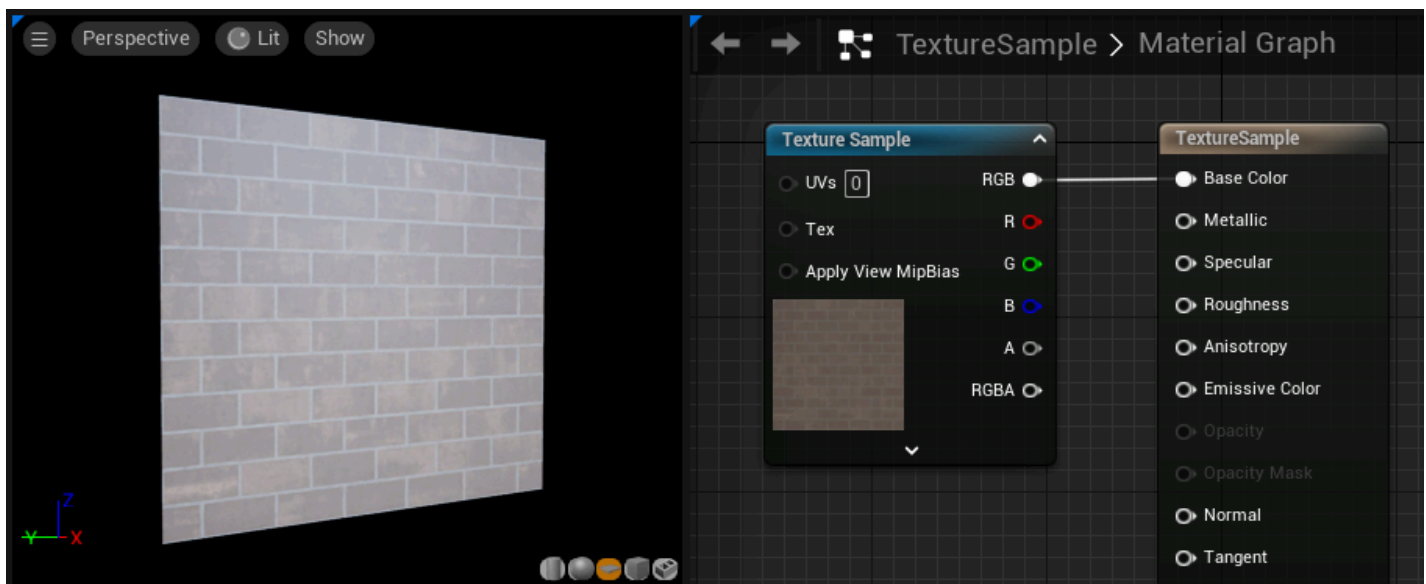
Item	Description
Properties	
Texture	The texture from the Content Browser that will be applied to this node.
Sampler Type	The type of data that will be output from the node.

TextureSample

The **TextureSample** expression outputs the color value(s) from a texture. This texture can be a regular Texture2D (including normal maps), a cubemap, or a movie texture.

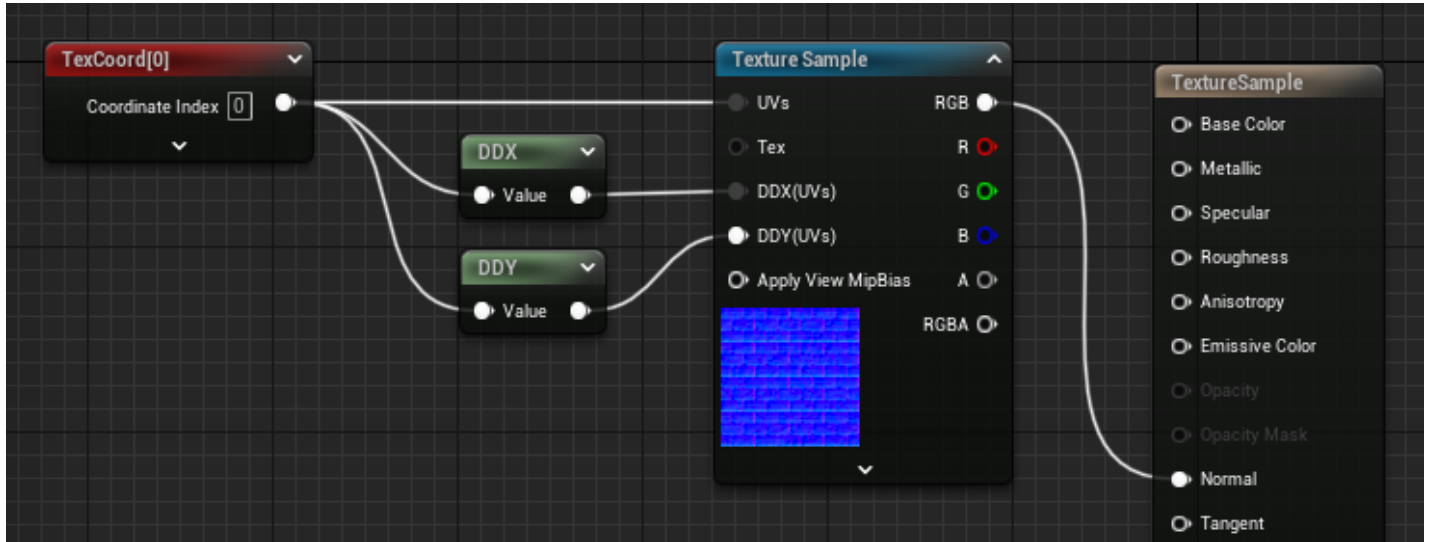
Property	Description
Texture	Specifies the texture sampled by the expression. To set the texture, first select a texture in the Content Browser . Then, select the Texture property in the expression's property window and click the 'Use Current Selection' button.
Sampler Type	The type of data that will be sampled and output from the node.
MipValueMode	Selects how to customize the sample's mip-level or derivatives from the default hardware computed. Affects the look and performance.
Inputs	
UVs	Takes in UV texture coordinates to use for the texture. If no values are input to the UVs, the texture coordinates of the mesh the Material is applied to are used. If the TextureSample represents a cubemap texture, the UVs coordinate must be a three-channel value rather than just a two-channel value.
DDX(UVs)	Takes the DDX of UVs input for anisotropic filtering. By default, the hardware automatically do it, and faster. But there is some cases where you may need to explicitly to set it avoiding the DDX Material expresion to avoid

Property	Description
	artifacts like the 2×2 pixels block artifact of the deferred decal. Only available with MipValueMode = Derivative .
DDY(UVs)	Takes the DDY of UVs input for anisotropic filtering as same as DDX(UVs) takes the DDX of the UVs input.
Outputs	
RGB	Outputs the three-channel RGB vector value of the color.
R	Outputs the red channel of the color.
G	Outputs the green channel of the color.
B	Outputs the blue channel of the color.
A	Outputs the alpha channel of the color. If a texture does not contain an alpha channel, connecting the 'alpha' channel to something, while not technically illegal, will always result in zero (black).



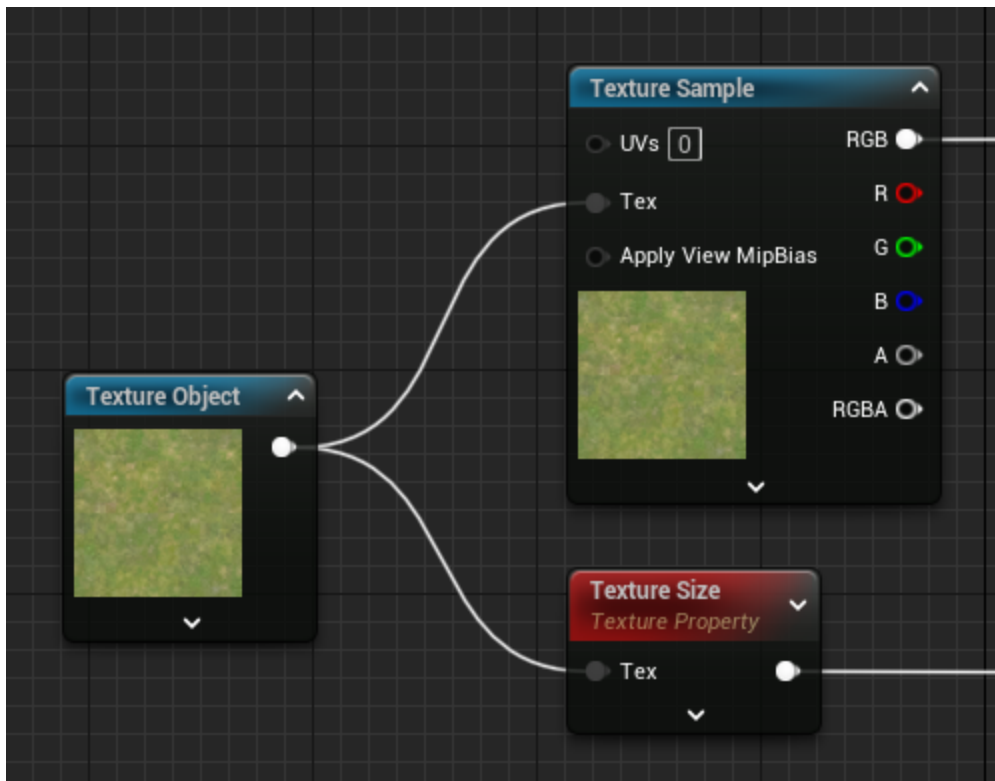
For a better understanding of the DDX(UVs) and DDY(UVs) inputs, you can set the **MipValueMode** to **Derivative** and configure the graph like the image below. Keep in mind that this has the exact same output as with **MipValueMode** = **None**, but is slower. The DDX(UVs)

and DDY(UVs) should only be used when you actually need to override the default hardware computation because computing those values by a different way than using the DDX/DDY material expression.



TextureProperty

The **TextureProperty** exposes a texture's property of your choice such as the texture's size or texel size.

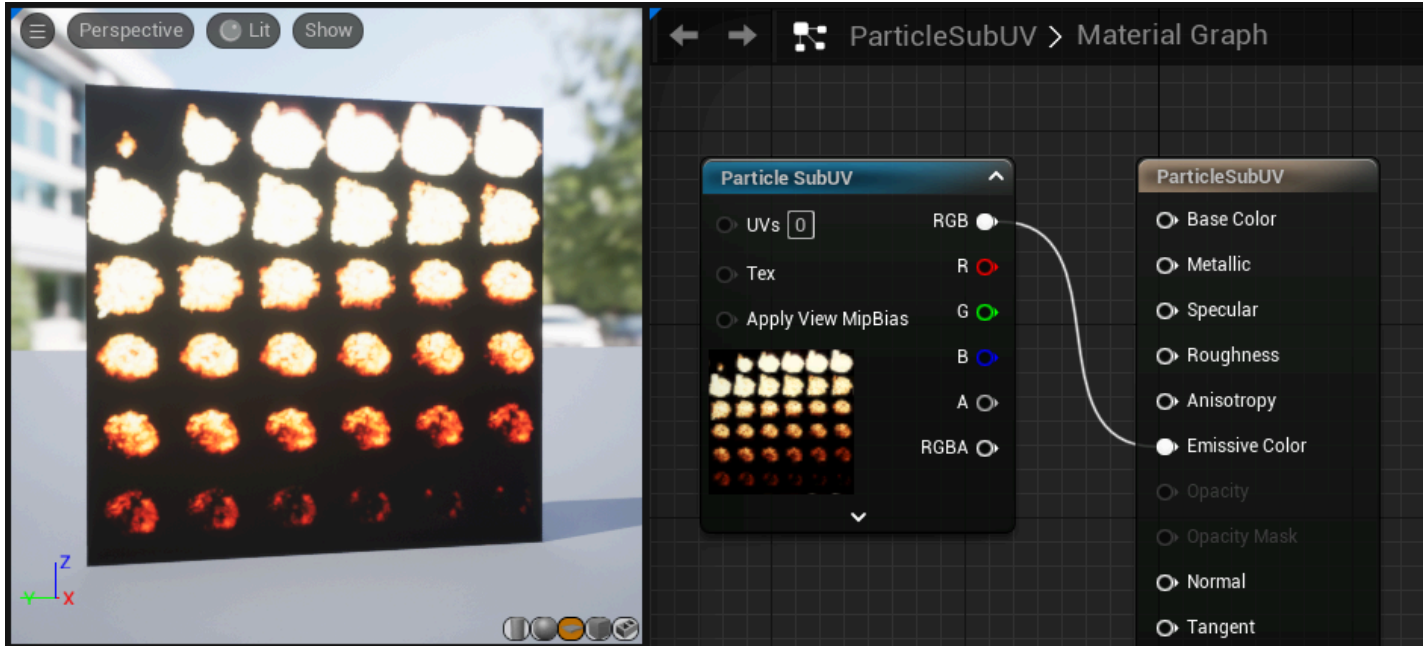


ParticleSubUV

The **ParticleSubUV** expression is used to render sub-images of a texture to a particle. ParticleSubUV is similar to a flipbook, except that ParticleSubUV allows the texture animation to be manipulated in Cascade.

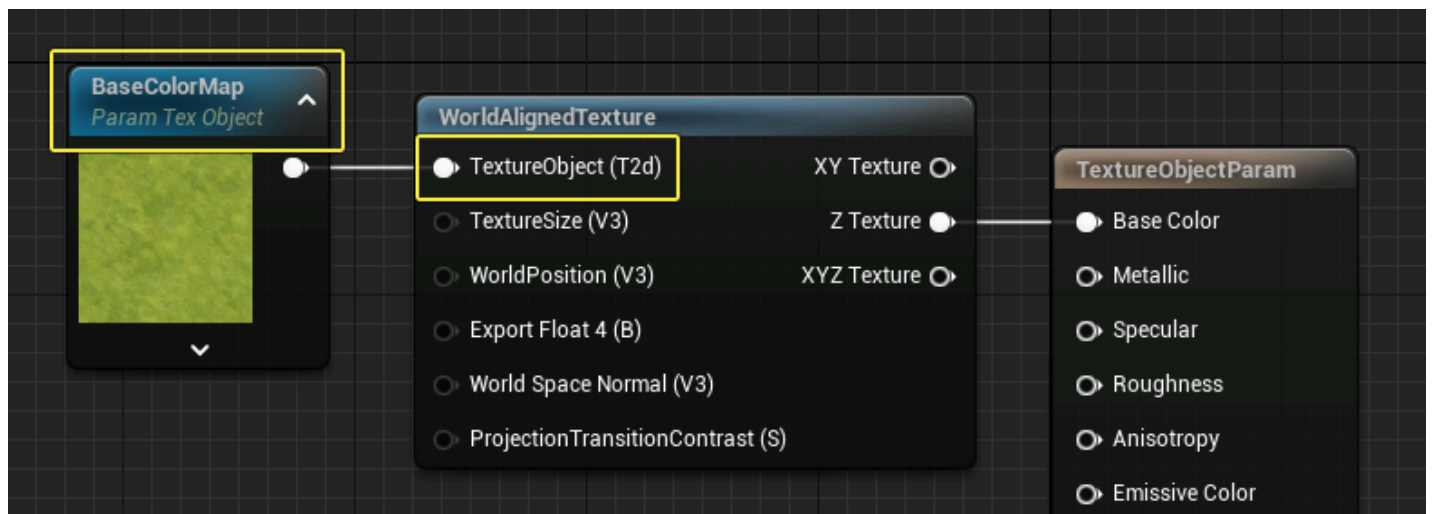
Item	Description
Properties	
Blend	Blends together each frame of the SubUV sprite layout, rather than instantly "popping" from one frame to the next.
Texture	Specifies the texture to use.
Sampler Type	The type of data that will be sampled and output from the node.
MipValueMode	Selects how to customize the sample's mip-level or derivatives from the default hardware computed. Affects the look and performance.
Inputs	
UVs	The UV input is ignored and does nothing.
Outputs	
RGB	Outputs the three-channel RGB vector value of the color.
R	Outputs the red channel of the color.
G	Outputs the green channel of the color.
B	Outputs the blue channel of the color.

Item	Description
A	Outputs the alpha channel of the color. If a texture does not contain an alpha channel, connecting the 'alpha' channel to something, while not technically illegal, will always result in zero (black).



TextureObjectParameter

The **TextureObjectParameter** expression defines a texture parameter and outputs the texture object. This expression is frequently used to pass texture parameters into a Material Function with texture inputs. Texture inputs on a Material Function node are not compatible with Float3 data from a TextureSample 2D node, so the Texture Object (T2d) is required. This node does not actually sample the texture, so it must be used in conjunction with a **TextureSample** node.

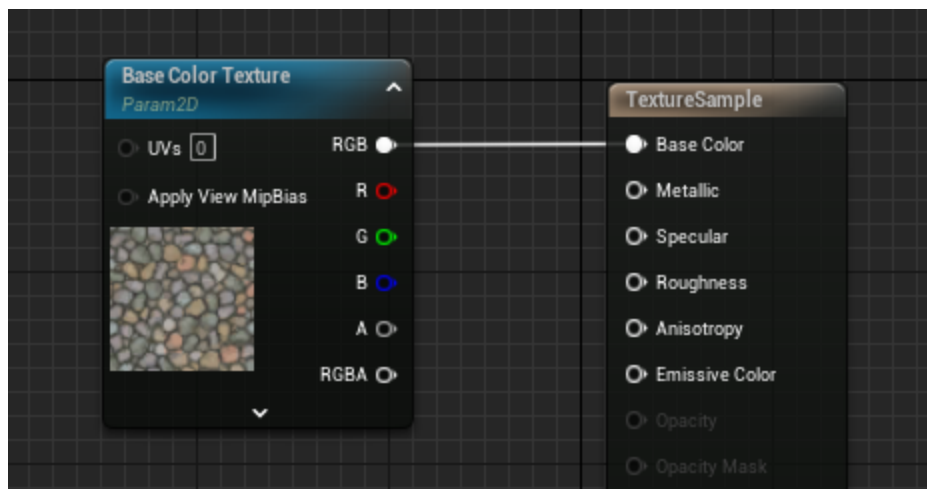


Property	Description
Parameter Name	Specifies the name used to identify the parameter in instances of the Material and through code.
Group	Provides a way to organize parameter names into groups, or categories, within a MaterialInstanceConstant. All parameters within a Material that have the same Group property name will be listed underneath that category in the Instance Editor.
Texture	Specifies the texture sampled by the expression.
Sampler Type	The type of data that will be sampled and output from the node.
MipValueMode	Selects how to customize the sample's mip-level or derivatives from the default hardware computed. Affects the look and performance.

This node is used with [MaterialFunctions](#).

TextureSampleParameter2D

The **TextureSampleParameter2D** expression is identical to the TextureSample except that it is a parameter that you can modify in Material Instances and through Blueprint or code.



Property	Description
Parameter Name	Specifies the name used to identify the parameter in instances of the Material and through code.
Group	Provides a way to organize parameter names into groups, or categories, within a MaterialInstanceConstant. All parameters within a Material that have the same Group property name will be listed underneath that category in the Instance Editor.
Texture	Specifies the texture sampled by the expression.
Sampler Type	The type of data that will be sampled and output from the node.
MipValueMode	Selects how to customize the sample's mip-level or derivatives from the default hardware computed. Affects the look and performance.
Inputs	
UVs	Takes in UV texture coordinates to use for the texture. If no values are input to the UVs, the texture coordinates of the mesh the material is applied to are used.
Outputs	
RGB	Outputs the three-channel RGB vector value of the color.

Property	Description
R	Outputs the red channel of the color.
G	Outputs the green channel of the color.
B	Outputs the blue channel of the color.
A	Outputs the alpha channel of the color. If a texture does not contain an alpha channel, connecting the 'alpha' channel to something, while not technically illegal, will always result in zero (black).

TextureSampleParameterSubUV

The **TextureSampleParameterSubUV** expression is identical to the [ParticleSubUV](#) except that it is a parameter that can be modified in instances of the material and through code.

Item	Description
Properties	
Parameter Name	Specifies the name used to identify the parameter in instances of the Material and through code.
Group	Provides a way to organize parameter names into groups, or categories, within a MaterialInstanceConstant. All parameters within a Material that have the same Group property name will be listed underneath that category in the Instance Editor.
Blend	Blends together each frame of the SubUV sprite layout, rather than instantly "popping" from one frame to the next.
Texture	Specifies the texture sampled by the expression.
Sampler Type	The type of data that will be sampled and output from the node.

Item	Description
MipValueMode	Selects how to customize the sample's mip-level or derivatives from the default hardware computed. Affects the look and performance.
Inputs	
UVs	The UV input is ignored and does nothing.
Outputs	
RGB	Outputs the three-channel RGB vector value of the color.
R	Outputs the red channel of the color.
G	Outputs the green channel of the color.
B	Outputs the blue channel of the color.
A	Outputs the alpha channel of the color. If a texture does not contain an alpha channel, connecting the 'alpha' channel to something, while not technically illegal, will always result in zero (black).

TextureSampleParameterCube

The **TextureSampleParameterCube** expression is identical to the TextureSample except that it only accepts cubemaps and it is a parameter that can be modified in instances of the material and through code.

Property	Description
Parameter Name	Specifies the name used to identify the parameter in instances of the Material and through code.
Group	Provides a way to organize parameter names into groups, or categories, within a MaterialInstanceConstant. All parameters within a Material that

Property	Description
	have the same Group property name will be listed underneath that category in the Instance Editor.
Texture	Specifies the texture sampled by the expression.
Sampler Type	The type of data that will be sampled and output from the node.
MipValueMode	Selects how to customize the sample's mip-level or derivatives from the default hardware computed. Affects the look and performance.
Inputs	
UVs	Takes in UV texture coordinates to use for the texture. If no values are input to the UVs, the texture coordinates of the mesh the Material is applied to are used. This input accepts a two-channel vector value.
Outputs	
RGB	Outputs the three-channel RGB vector value of the color.
R	Outputs the red channel of the color.
G	Outputs the green channel of the color.
B	Outputs the blue channel of the color.
A	Outputs the alpha channel of the color. If a texture does not contain an alpha channel, connecting the 'alpha' channel to something, while not technically invalid, will always result in zero (black).