

- Developer
- / Documentation
- / Unreal Engine ▾
- / Unreal Engine 5.4 Documentation
- / Designing Visuals, Rendering, and Graphics
- / Materials
- / Material Expressions Reference
- / Material Attributes Expressions

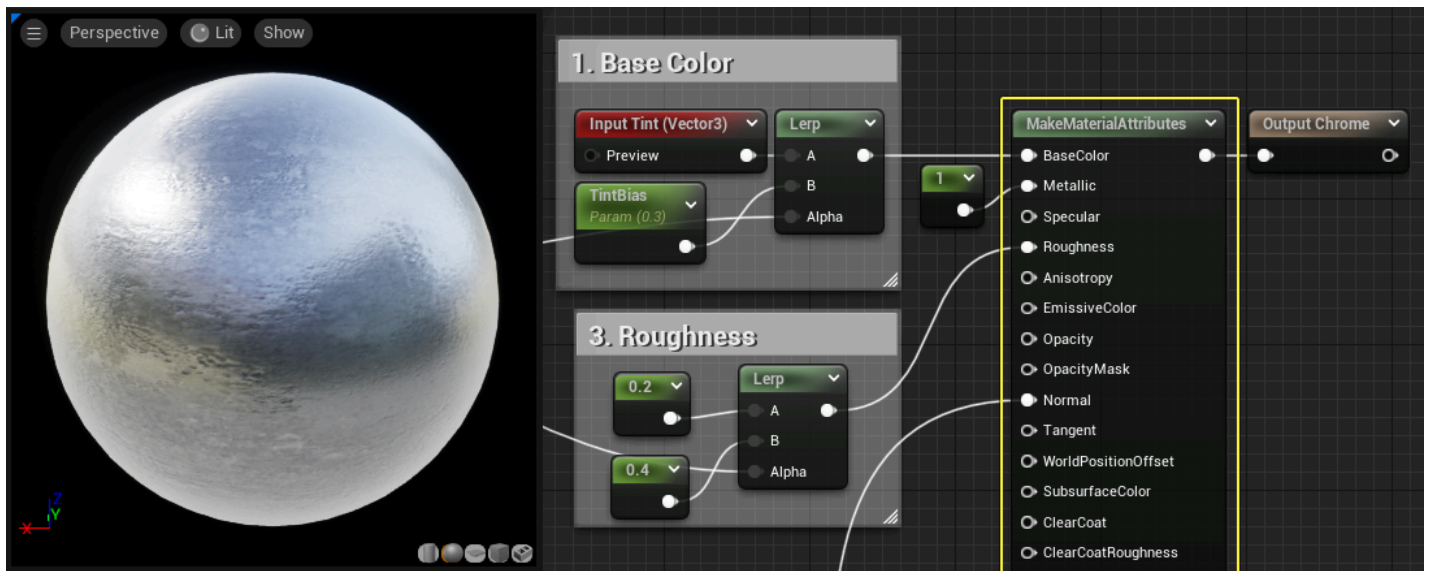
# Material Attributes Expressions

These expression nodes serve as a way to separate or combine various Material attributes, which is particularly useful when creating Layered Materials.

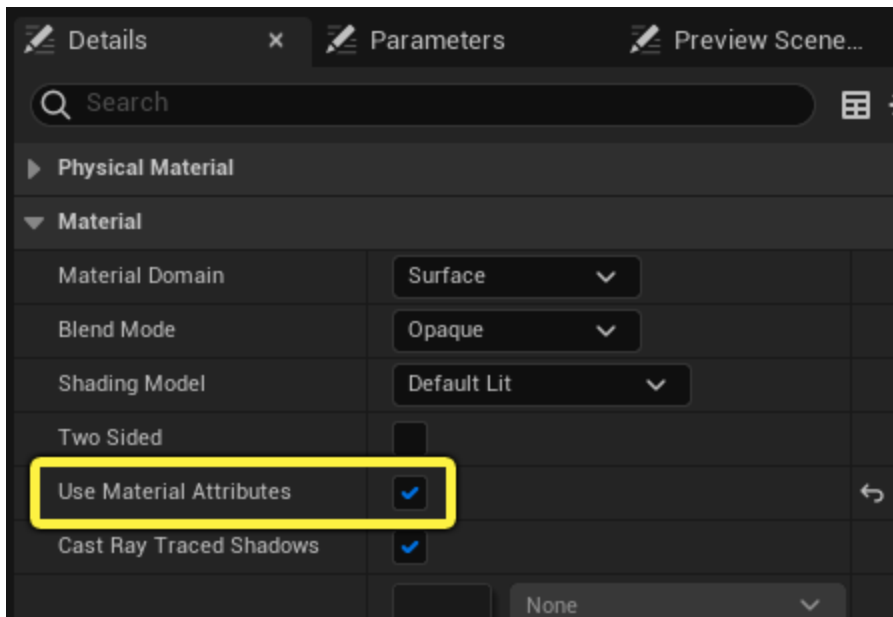


## Make Material Attributes

The **Make Material Attributes** node allows you to define any of the standard Material attributes found on the [Main Material Node](#), bundle them together, and pass them along through a single output. This is useful when creating [Layered Materials](#), as it allows you to define an entire Material within a Material Function and then utilize that data in a parent Material. You can also use it in complex Material setups in which you want to define multiple Material types and blend them together, all within one Material asset.



When connecting a Make Material Attributes node to a Material, you must make sure that the Use Material Attributes option is set to true (checked) in the Material properties. This will collapse the Main Material Node down to a single input, which accepts the incoming data from a Make Material Attributes node.

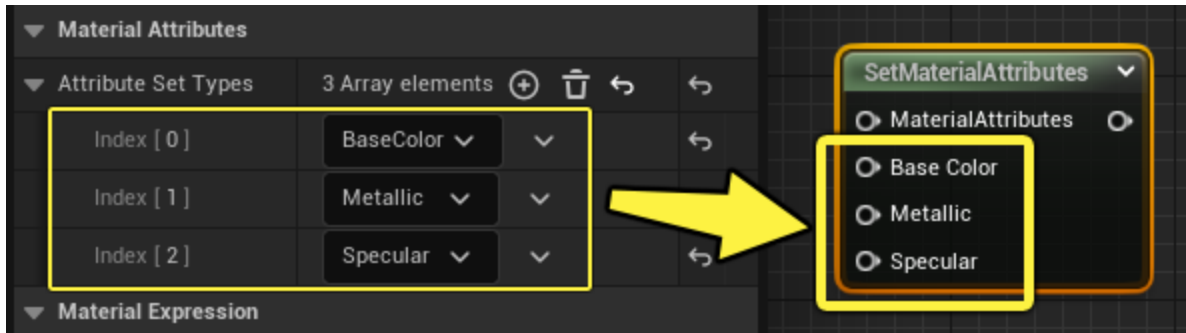


Make Material Attributes is superseded by the [Set Material Attributes](#) node, and in most cases you should use that instead.

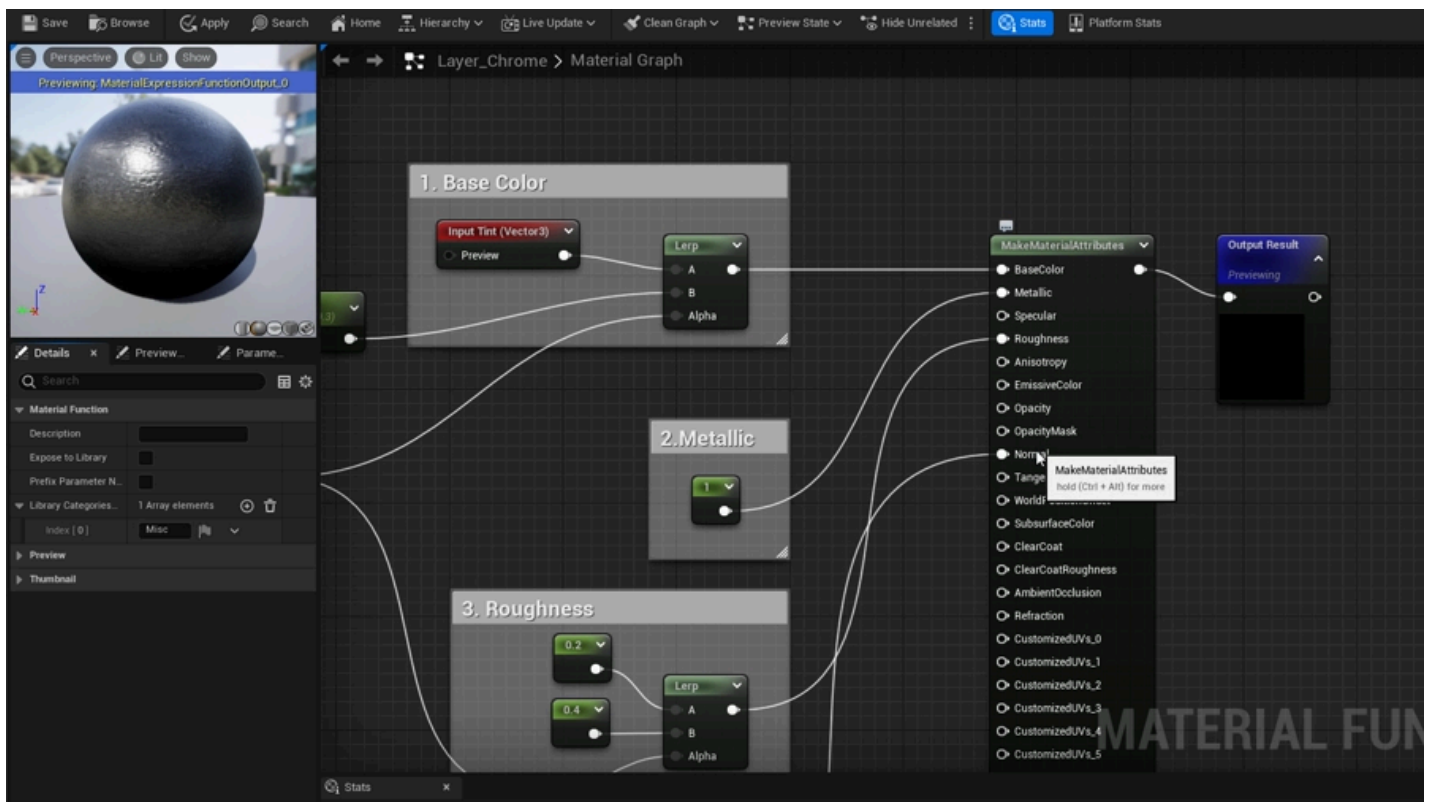
## Set Material Attributes

The **Set Material Attributes** expression serves the same purpose as Make Material Attributes. It allows you to define a set of Material Attributes, and pass them along as bundled data through a single wire.

Unlike the Make Material Attributes node, which displays all standard Material Attributes by default, the inputs that appear on the Set Material Attributes node are user-defined in the **Details panel**. This allows you to incorporate only the Material Attributes that you need.



For example, the Material Function shown below only requires four attributes — **Base Color**, **Metallic**, **Roughness**, and **Normal**. Using the Set Material Attributes expression instead of Make Material Attributes takes up less space and results in a more concise Material Graph.



In most cases, Set Material Attributes is preferred to Make Material Attributes.

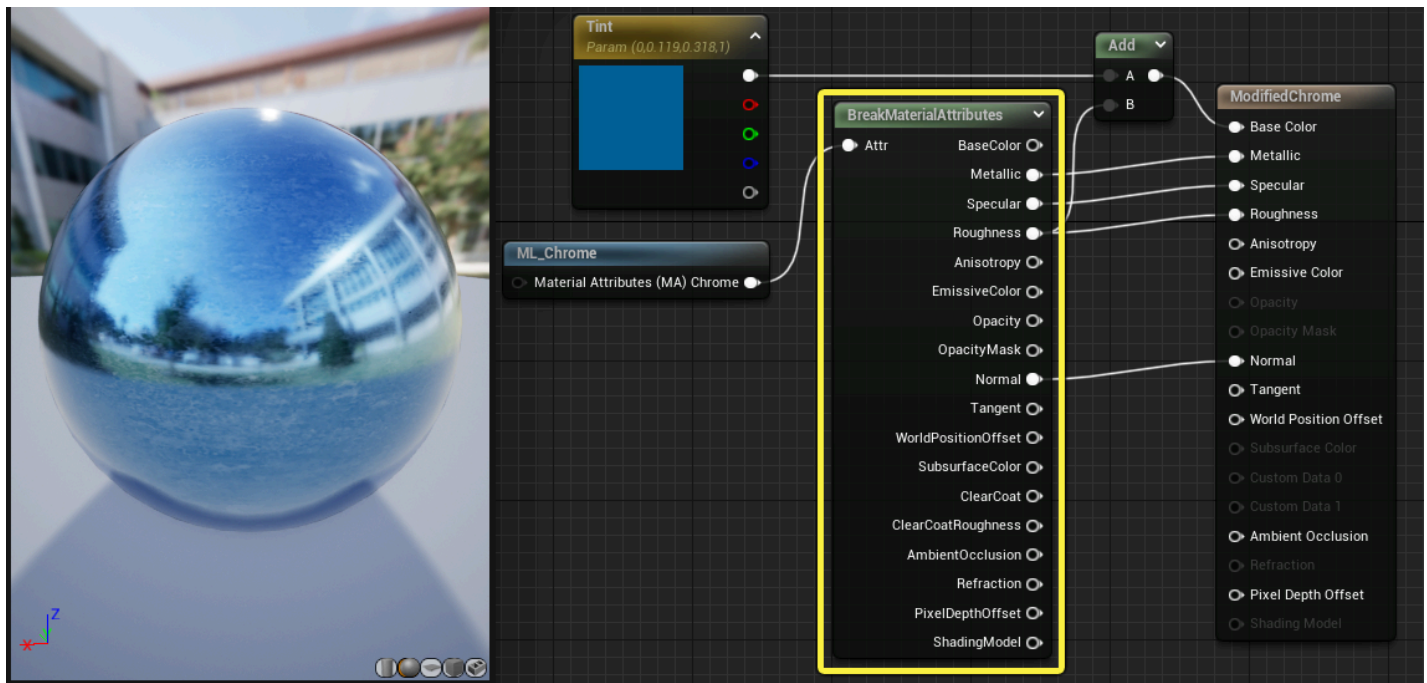


The only drawback is that the node is not context sensitive, meaning the user must know which attributes they need to add to the array for various combinations of Blend Mode and Shading Model. If you are unsure, you can use the Make Material Attributes node, which contains all necessary inputs.

## Break Material Attributes

The **Break Material Attributes** expression splits up an incoming set of Material Attributes, and outputs each individual attribute through a separate pin.

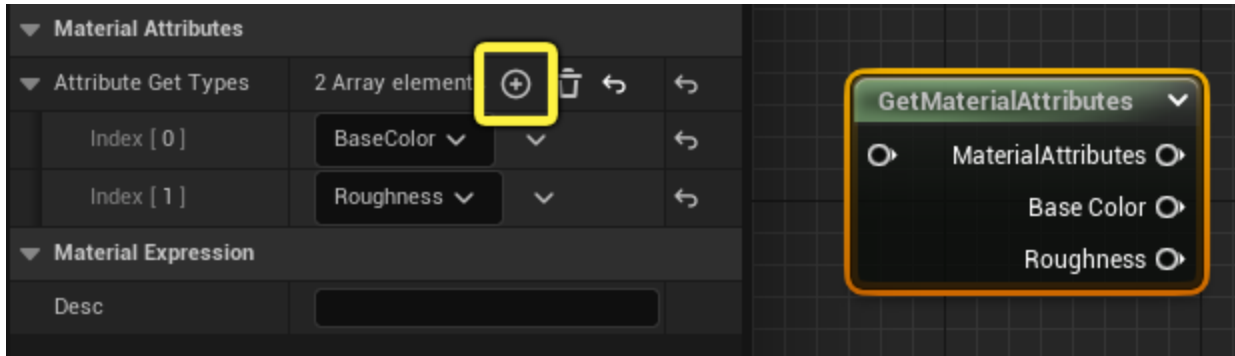
This is useful when creating [Layered Materials](#), because it allows you to access the individual attributes within a Material Layer function. This enables you choose which attributes you want to plug into the Main Material node, and gives you the ability to selectively modify the attributes with additional logic in the Material Graph.



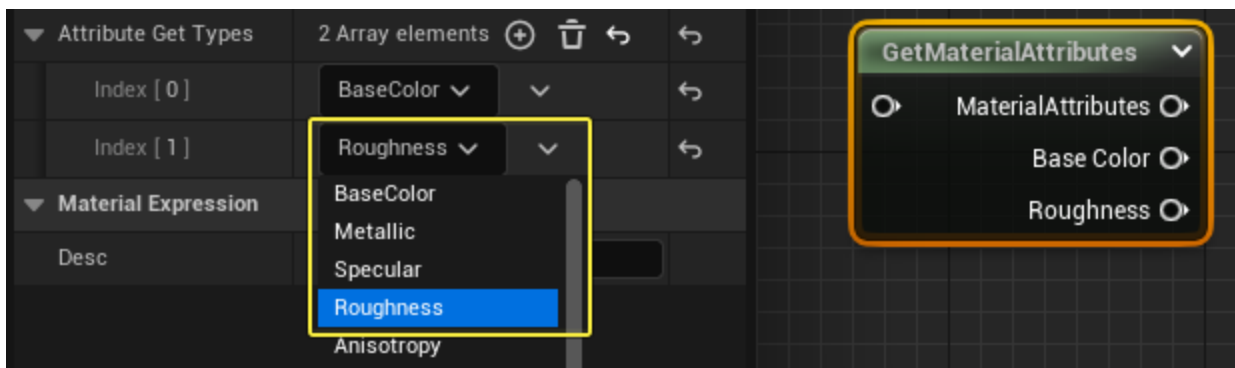
In the example above, Break Material Attributes is used to split up the attributes from a generic chrome Material Layer Function. The Metallic, Specular, Roughness and Normal attributes are passed into the Main Material Node unchanged, but the Base Color attribute is not used. Instead, a Vector 3 parameter is added to the Roughness texture and this result is passed into the Base Color input.

# Get Material Attributes

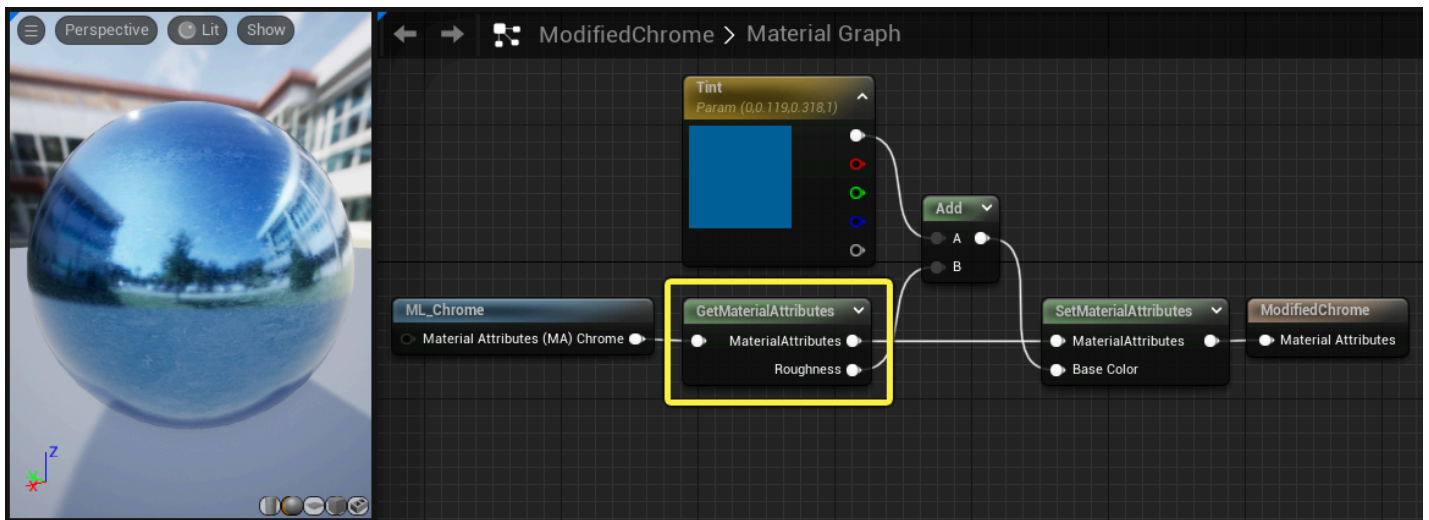
**Get Material Attributes** performs the same task as Break Material Attributes, but offers significant workflow benefits. Instead of fully breaking apart the incoming Material Attributes, Get Material Attributes allows you to pull out only the attributes you need. Select the node and then click the **Add Element** icon in the **Details Panel** to add output pins.



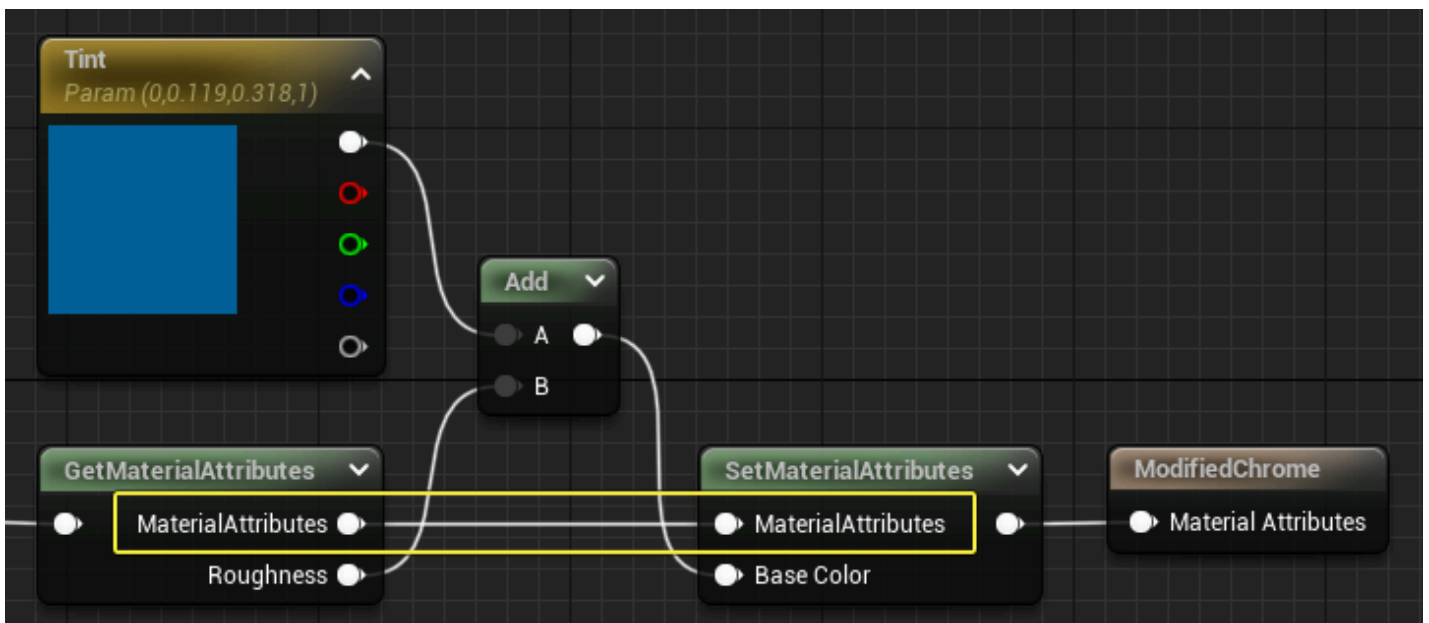
You can use the dropdown menus to define which Material Attribute is targeted by each array element.



The Material shown below recreates the tinted chrome example from the [Break Material Attributes](#) section, but uses the Get and Set Material Attributes expressions instead. Note that the graph is cleaner and requires fewer wires.



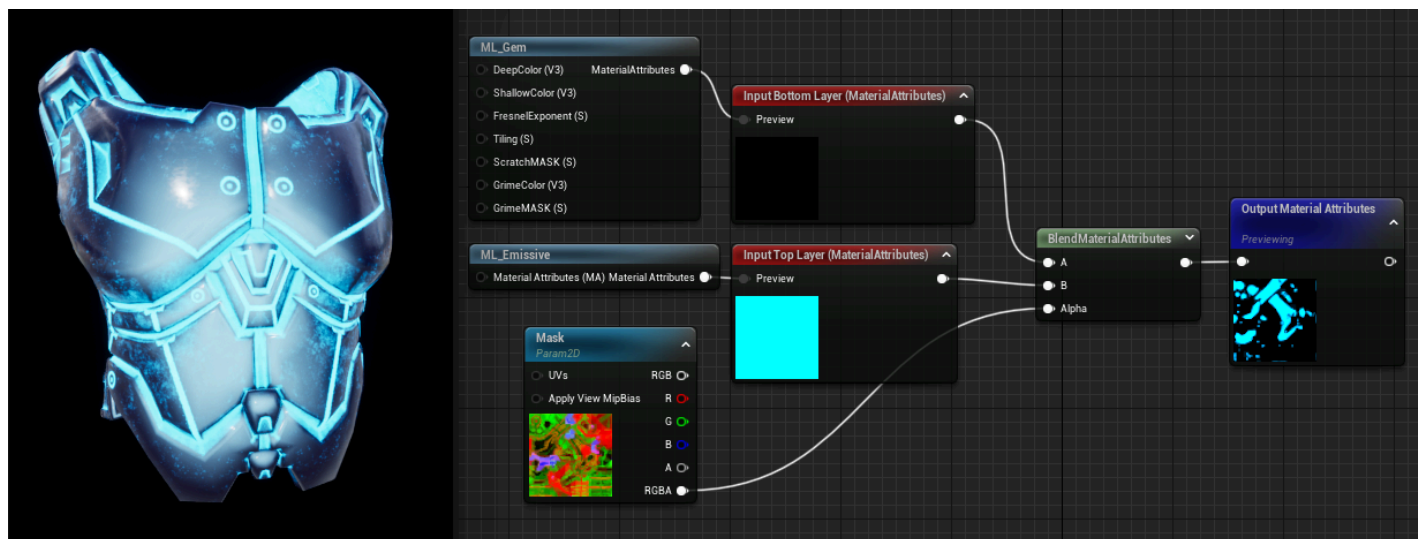
In this case, only the Roughness data is broken out of the Material Attributes. A tint parameter is then added to the Roughness map to create some surface variation and the result is passed into Base Color with a [Set Material Attributes](#) node. The rest of the Material Attributes simply pass through unmodified.



## Blend Material Attributes

The **Blend Material Attributes** expression takes two sets of Material Attributes, and blends them together using per-pixel operations defined in the Alpha input.

For example, this Material Layer Blend takes a top and bottom set of Material Attributes, and blends between them using a mask texture.



*Click image to enlarge.*