

Introduction to Shader development

h_da WS2020/21

Paul Nasdalack

info@paul-nasdalack.com

 [@littleBugHunter](https://twitter.com/littleBugHunter)

Congrats!

- Texture mapping
- UV scrolling
- Displacement mapping
- Tinting

You can go pretty much anywhere from here



So what now?

TRANSPARENCY!

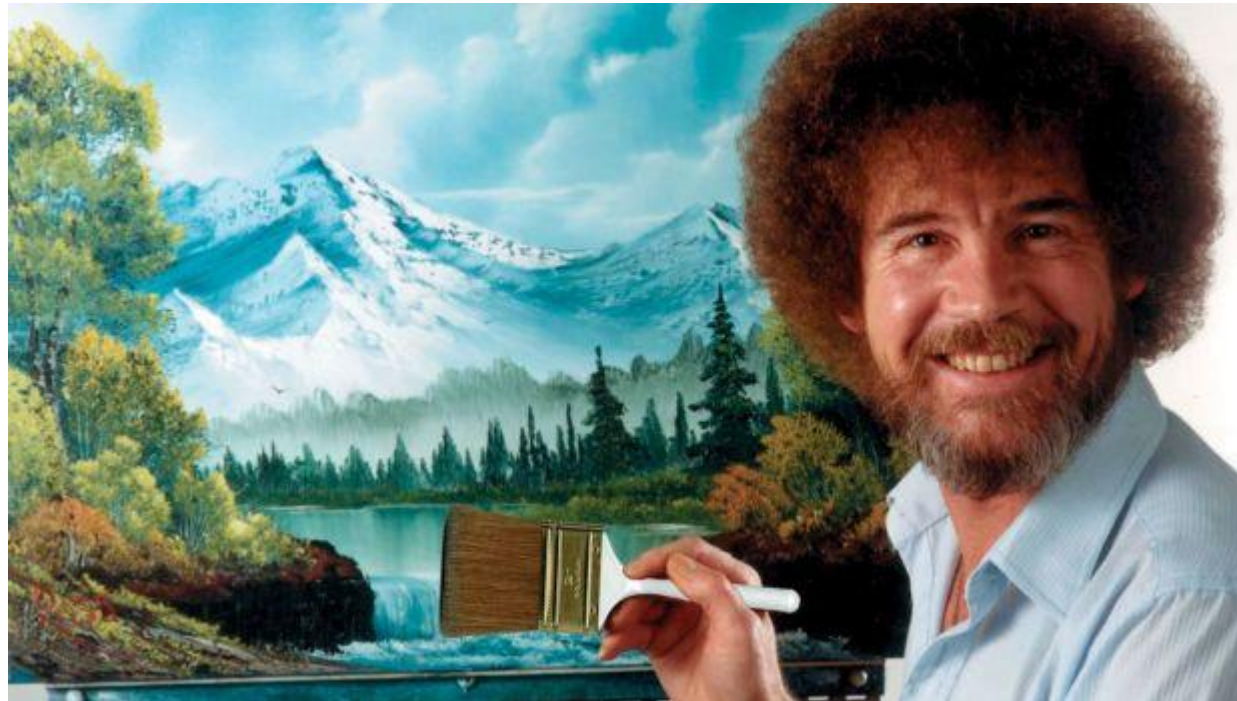


How does the GPU Render Objects

- What is in front and what is behind?
 - Painters Algorithm
 - Zbuffer Compare



Painters Algorithm

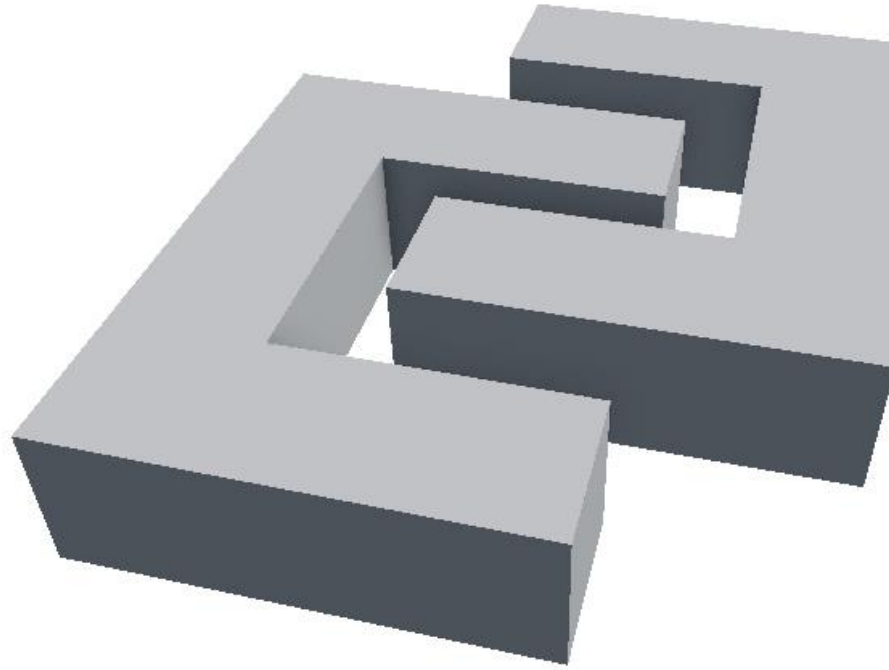


Painters Algorithm

- Draw Objects from back to front
 - Like Photoshop Layers



Painters Algorithm

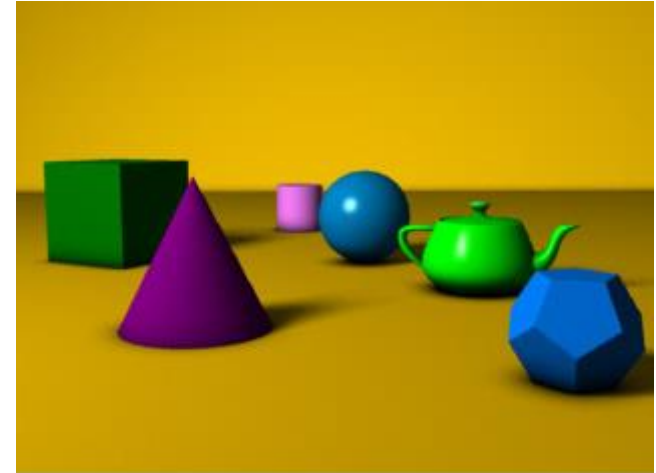


Painters Algorithm

- We have to cut objects in pieces (at runtime?!)
- Renders a ton of stuff, that is later overdrawn
- Works ok for simple scenes
- We need something better

ZBuffer

- Distance from the camera
- Stored in a second Texture
- Rasterizer checks each Pixels distance
- Resolved per pixel, not per object



ZBuffer

- Works much better for complex scenes
- Objects can be rendered front to back to avoid overdraw
- Soo... YAY?

Back to Transparency!

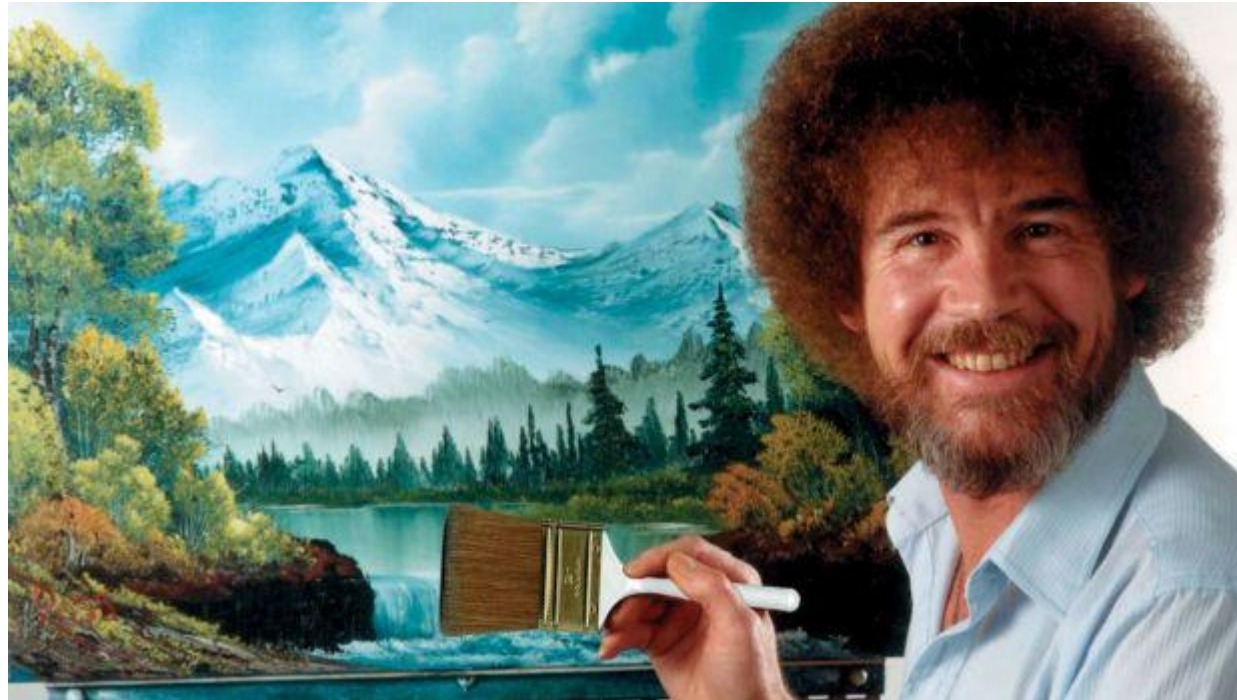


We have a problem

- We render front to back
- We discard all pixels, that are behind our object
- There is nothing behind the transparent object

We have a problem

- BACK TO PAINTERS ALGORITHM!



How Objects are Rendered

- Render all opaque objects using Z-Buffer
- Sort transparent objects back to front
- Render transparent objects using Painters Algorithm
 - Also check the Z-Buffer, but don't write to it

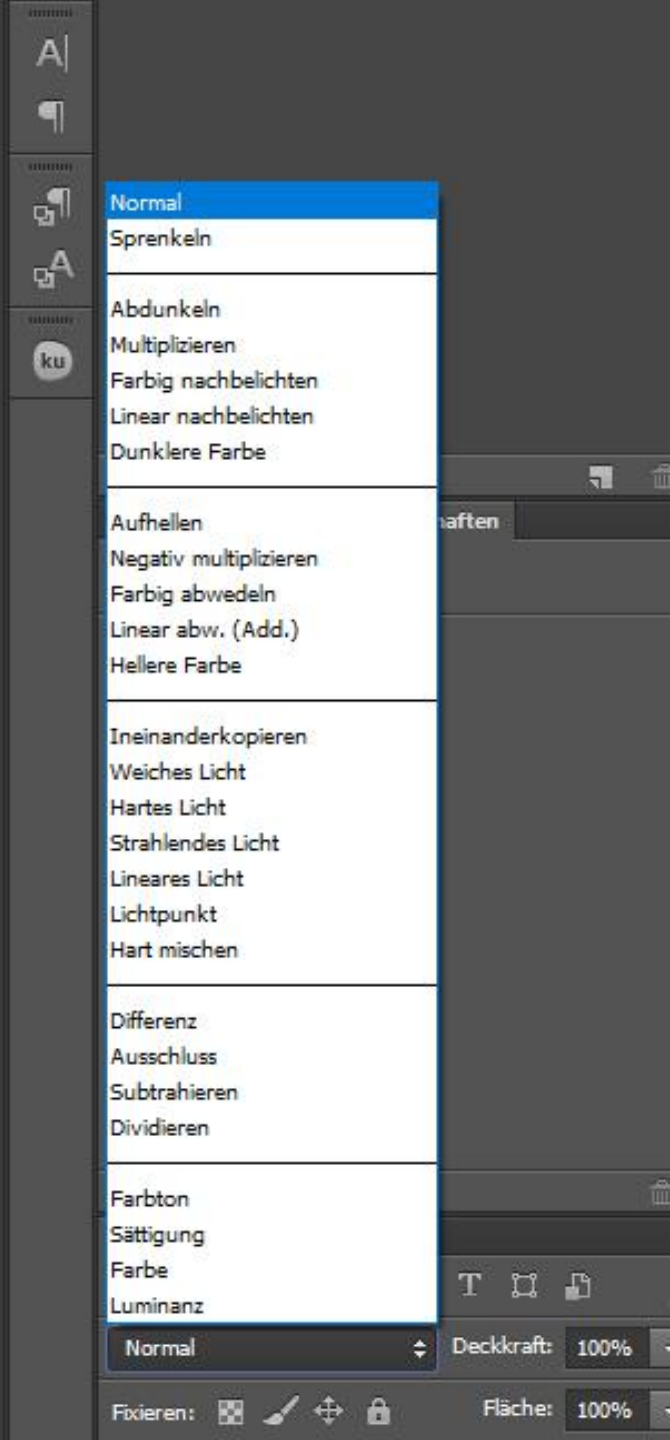
(how) Will it BLEND?



Blend Modes

- How to blend the object with the background
- Kind of like layer blend modes
- Always follow this formular:
(You can look them up online)

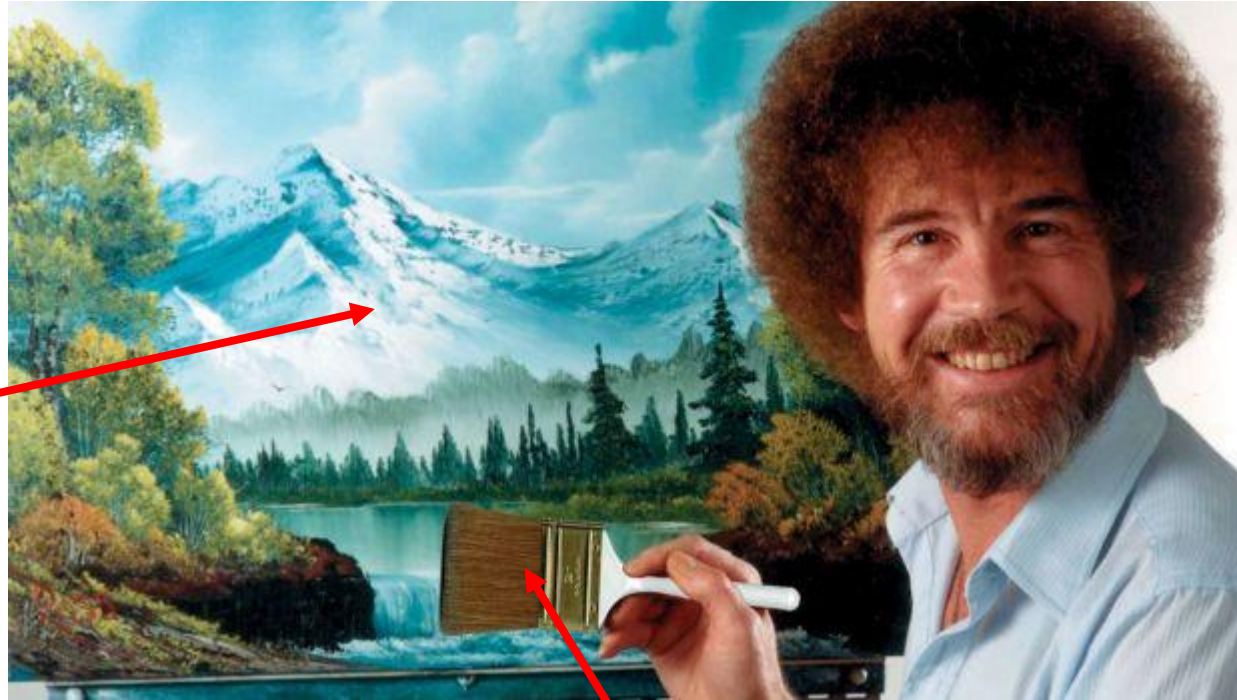
$\text{SrcColor} * \text{SrcBlend} + \text{DstColor} * \text{DstBlend}$



Blend Modes

$\text{SrcColor} * \text{SrcBlend} + \text{DstColor} * \text{DstBlend}$

DstColor



SrcColor

Blend Modes

`SrcColor`*`SrcBlend` + `DstColor`*`DstBlend`

One	The value of one - use this to let either the source or the destination color come through fully.
Zero	The value zero - use this to remove either the source or the destination values.
SrcColor	The value of this stage is multiplied by the source color value.
SrcAlpha	The value of this stage is multiplied by the source alpha value.
DstColor	The value of this stage is multiplied by frame buffer source color value.
DstAlpha	The value of this stage is multiplied by frame buffer source alpha value.
OneMinusSrcColor	The value of this stage is multiplied by (1 - source color).
OneMinusSrcAlpha	The value of this stage is multiplied by (1 - source alpha).
OneMinusDstColor	The value of this stage is multiplied by (1 - destination color).
OneMinusDstAlpha	The value of this stage is multiplied by (1 - destination alpha).

Blend Modes

`SrcColor`*`SrcBlend` + `DstColor`*`DstBlend`

One	The value of one - use this to let either the source or the destination color come through fully.
Zero	The value zero - use this to remove either the source or the destination values.
SrcColor	The value of this stage is multiplied by the source color value.
SrcAlpha	The value of this stage is multiplied by the source alpha value.
DstColor	The value of this stage is multiplied by frame buffer source color value.
DstAlpha	The value of this stage is multiplied by frame buffer source alpha value.
OneMinusSrcColor	The value of this stage is multiplied by (1 - source color).
OneMinusSrcAlpha	The value of this stage is multiplied by (1 - source alpha).
OneMinusDstColor	The value of this stage is multiplied by (1 - destination color).
OneMinusDstAlpha	The value of this stage is multiplied by (1 - destination alpha).

Blend Modes

`SrcColor*One + DstColor*One`

One	The value of one - use this to let either the source or the destination color come through fully.
Zero	The value zero - use this to remove either the source or the destination values.
SrcColor	The value of this stage is multiplied by the source color value.
SrcAlpha	The value of this stage is multiplied by the source alpha value.
DstColor	The value of this stage is multiplied by frame buffer source color value.
DstAlpha	The value of this stage is multiplied by frame buffer source alpha value.
OneMinusSrcColor	The value of this stage is multiplied by (1 - source color).
OneMinusSrcAlpha	The value of this stage is multiplied by (1 - source alpha).
OneMinusDstColor	The value of this stage is multiplied by (1 - destination color).
OneMinusDstAlpha	The value of this stage is multiplied by (1 - destination alpha).

Blend Modes

`SrcColor*1 + DstColor*1`

One	The value of one - use this to let either the source or the destination color come through fully.
Zero	The value zero - use this to remove either the source or the destination values.
SrcColor	The value of this stage is multiplied by the source color value.
SrcAlpha	The value of this stage is multiplied by the source alpha value.
DstColor	The value of this stage is multiplied by frame buffer source color value.
DstAlpha	The value of this stage is multiplied by frame buffer source alpha value.
OneMinusSrcColor	The value of this stage is multiplied by (1 - source color).
OneMinusSrcAlpha	The value of this stage is multiplied by (1 - source alpha).
OneMinusDstColor	The value of this stage is multiplied by (1 - destination color).
OneMinusDstAlpha	The value of this stage is multiplied by (1 - destination alpha).

Blend Modes

SrcColor + DstColor

One	The value of one - use this to let either the source or the destination color come through fully.
Zero	The value zero - use this to remove either the source or the destination values.
SrcColor	The value of this stage is multiplied by the source color value.
SrcAlpha	The value of this stage is multiplied by the source alpha value.
DstColor	The value of this stage is multiplied by frame buffer source color value.
DstAlpha	The value of this stage is multiplied by frame buffer source alpha value.
OneMinusSrcColor	The value of this stage is multiplied by (1 - source color).
OneMinusSrcAlpha	The value of this stage is multiplied by (1 - source alpha).
OneMinusDstColor	The value of this stage is multiplied by (1 - destination color).
OneMinusDstAlpha	The value of this stage is multiplied by (1 - destination alpha).

Blend Modes

`SrcColor + DstColor = Additive Blending`

One	The value of one - use this to let either the source or the destination color come through fully.
Zero	The value zero - use this to remove either the source or the destination values.
SrcColor	The value of this stage is multiplied by the source color value.
SrcAlpha	The value of this stage is multiplied by the source alpha value.
DstColor	The value of this stage is multiplied by frame buffer source color value.
DstAlpha	The value of this stage is multiplied by frame buffer source alpha value.
OneMinusSrcColor	The value of this stage is multiplied by (1 - source color).
OneMinusSrcAlpha	The value of this stage is multiplied by (1 - source alpha).
OneMinusDstColor	The value of this stage is multiplied by (1 - destination color).
OneMinusDstAlpha	The value of this stage is multiplied by (1 - destination alpha).

Multiplicative Blending (NewColor*OldColor)

`SrcColor`*`SrcBlend` + `DstColor`*`DstBlend`

One	The value of one - use this to let either the source or the destination color come through fully.
Zero	The value zero - use this to remove either the source or the destination values.
SrcColor	The value of this stage is multiplied by the source color value.
SrcAlpha	The value of this stage is multiplied by the source alpha value.
DstColor	The value of this stage is multiplied by frame buffer source color value.
DstAlpha	The value of this stage is multiplied by frame buffer source alpha value.
OneMinusSrcColor	The value of this stage is multiplied by (1 - source color).
OneMinusSrcAlpha	The value of this stage is multiplied by (1 - source alpha).
OneMinusDstColor	The value of this stage is multiplied by (1 - destination color).
OneMinusDstAlpha	The value of this stage is multiplied by (1 - destination alpha).

Multiplicative Blending (NewColor*OldColor)

`SrcColor`*`SrcBlend` + `DstColor`*`DstBlend`

One	The value of one - use this to let either the source or the destination color come through fully.
Zero	The value zero - use this to remove either the source or the destination values.
<code>SrcColor</code>	The value of this stage is multiplied by the source color value.
<code>SrcAlpha</code>	The value of this stage is multiplied by the source alpha value.
<code>DstColor</code>	The value of this stage is multiplied by frame buffer source color value.
<code>DstAlpha</code>	The value of this stage is multiplied by frame buffer source alpha value.
<code>OneMinusSrcColor</code>	The value of this stage is multiplied by (1 - source color).
<code>OneMinusSrcAlpha</code>	The value of this stage is multiplied by (1 - source alpha).
<code>OneMinusDstColor</code>	The value of this stage is multiplied by (1 - destination color).
<code>OneMinusDstAlpha</code>	The value of this stage is multiplied by (1 - destination alpha).

Multiplicative Blending (NewColor*OldColor)

SrcColor*DstColor + DstColor*Zero

One	The value of one - use this to let either the source or the destination color come through fully.
Zero	The value zero - use this to remove either the source or the destination values.
SrcColor	The value of this stage is multiplied by the source color value.
SrcAlpha	The value of this stage is multiplied by the source alpha value.
DstColor	The value of this stage is multiplied by frame buffer source color value.
DstAlpha	The value of this stage is multiplied by frame buffer source alpha value.
OneMinusSrcColor	The value of this stage is multiplied by (1 - source color).
OneMinusSrcAlpha	The value of this stage is multiplied by (1 - source alpha).
OneMinusDstColor	The value of this stage is multiplied by (1 - destination color).
OneMinusDstAlpha	The value of this stage is multiplied by (1 - destination alpha).

Multiplicative Blending (NewColor*OldColor)

`SrcColor*DstColor + DstColor*0`

One	The value of one - use this to let either the source or the destination color come through fully.
Zero	The value zero - use this to remove either the source or the destination values.
SrcColor	The value of this stage is multiplied by the source color value.
SrcAlpha	The value of this stage is multiplied by the source alpha value.
DstColor	The value of this stage is multiplied by frame buffer source color value.
DstAlpha	The value of this stage is multiplied by frame buffer source alpha value.
OneMinusSrcColor	The value of this stage is multiplied by (1 - source color).
OneMinusSrcAlpha	The value of this stage is multiplied by (1 - source alpha).
OneMinusDstColor	The value of this stage is multiplied by (1 - destination color).
OneMinusDstAlpha	The value of this stage is multiplied by (1 - destination alpha).

Multiplicative Blending (NewColor*OldColor)

SrcColor*DstColor

One	The value of one - use this to let either the source or the destination color come through fully.
Zero	The value zero - use this to remove either the source or the destination values.
SrcColor	The value of this stage is multiplied by the source color value.
SrcAlpha	The value of this stage is multiplied by the source alpha value.
DstColor	The value of this stage is multiplied by frame buffer source color value.
DstAlpha	The value of this stage is multiplied by frame buffer source alpha value.
OneMinusSrcColor	The value of this stage is multiplied by (1 - source color).
OneMinusSrcAlpha	The value of this stage is multiplied by (1 - source alpha).
OneMinusDstColor	The value of this stage is multiplied by (1 - destination color).
OneMinusDstAlpha	The value of this stage is multiplied by (1 - destination alpha).

Configuration

- Global states we can set (like in the old days)
- Usually handled by the Engine
- In UE4 and Unity ShaderGraph you set the States in the Material Properties
- In Unity you can use the Meta language, that wraps the shader
- In your own Engine you'll directly call the GPU through DirectX or OpenGL