

Developer

/ Documentation

/ Unreal Engine ▾

/ Unreal Engine 5.4 Documentation

/ Testing and Optimizing Your Content

/ Using Oodle

/ Oodle Texture

Oodle Texture

An overview of using the Oodle Texture encoding solution to optimize textures in your project.

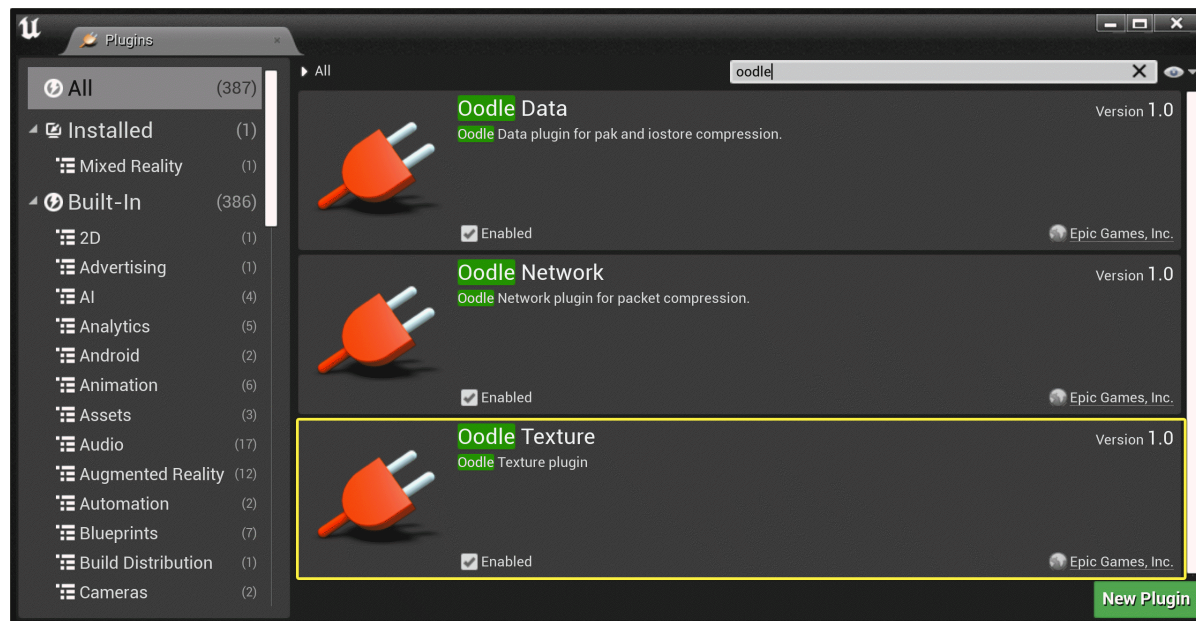


Oodle Texture provides fast, high quality encoding of textures to the various BCn/DXTn formats. Once Oodle Texture is configured it will operate automatically in the background. You can set Oodle Texture globally, and then define it more specifically for LOD groups and individual textures.

! Oodle Texture does not encode ASTC or other mobile formats.

Enabling Oodle Texture

The plugin for Oodle Texture is enabled by default in Unreal Engine.




In addition to the plugin, Oodle Texture requires a setting in the `DefaultEngine.ini` file.

```
1
2 \Engine\Config\DefaultEngine.ini
3 [AlternateTextureCompression]
4 TextureCompressionFormat="TextureFormatOodle"
5 TextureFormatPrefix="OODLE_"
```

```
6 bEnableInEditor=True
7
```

 Copy full snippet

 As Oodle Texture is enabled by default in **Unreal Engine**, these lines should already be present in your `BaseEngine.ini` file.

We strongly recommend leaving `bEnableInEditor=true` to maintain consistent behavior between the editor and packaged builds. If set to `false`, artists reviewing encoding results in the Editor will see different results than the cook system produces. You can verify Oodle Texture is enabled by examining the log:

```
LogTextureFormatOodle: Display: Oodle Texture 2.9.0 init RDO On with DefaultRDOLambda=30
```

When Oodle is used for a given texture, the format will contain the prefix `OODLE_`:

```
LogTexture: Display: Building textures: test (OODLE_AutoDXT, 256X256)
```

Key Concepts for Oodle Texture

There are two concepts you need to understand in order to make use of Oodle Texture: **RDO (Rate Distortion Optimization)** and **Lambda**.

Understanding RDO

RDO is a term that refers to trading quality (distortion) for size (rate). For texture encoding, this sounds odd — DXTn/BCn textures do not vary in size with quality, they are a fixed size based on the format, resolution, and mip count.

Oodle Texture optionally expose a way to manage the resulting encoded texture data so that when a uasset containing a texture is compressed for distribution via the IOStore / .pak file system, it compresses smaller. Thus, RDO in Oodle Texture only reduces distribution sizes.

Additionally, it is tuned to work with the Kraken compression format. Refer to [Oodle Data](#) for more information.

Understanding Lambda

The parameter that determines how much distortion is introduced, and thus how much smaller the resulting file is, is referred to as lambda.

Lambda can be set between 0 and 100, with lower numbers representing lower distortion, and therefore higher quality results. A lambda value of around 30 still produces high quality results. A lambda value of 0 disables RDO entirely, resulting in the theoretical best quality. However, even when seeking best quality, we recommend using a lambda value of 1 as the cost / benefit ratio is still very good, resulting in very little distortion for reasonable distribution size gains.

Generally speaking, we expect lambda to be set globally and not often overridden. Determining a value appropriate for your project will be a collaborative effort based on distribution size needs. It is best to have your global lambda be your highest value (lowest quality) one, and selectively set higher quality / lower lambda on LOD groups or specific textures, as needed.



Textures other than diffuse / albedo maps will likely require a lower lambda (usually 5-20), especially normal maps, as distortion that is not visible to the naked eye can be more noticeable with textures like specular highlights.

Configuring Oodle Texture

Oodle Texture is primarily configured using the `DefaultEngine.ini` file, but also exposes lambda on texture LOD groups and on a per-texture basis.

Global Configuration

The `TextureFormatOodle` section in the `DefaultEngine.ini` file contains the global settings for Oodle Texture.

```
1
2 \Engine\Config\DefaultEngine.ini
3 [TextureFormatOodle]
4 DefaultRDOLambda=30
5 GlobalLambdaMultiplier=1.0
6 bForceAllBC23ToBC7=False
7 bForceRDOff=False
8 bDebugColor=False
9
```

 Copy full snippet

Setting

Definition

`DefaultRDOLambda`

This is the RDO lambda that will be used if not set anywhere else.

`GlobalLambdaMultiplier`

This scales the lambda passed to the encoder. It is provided as a last-minute control to manage distribution sizes without having to chase down individual textures or LOD groups. The value is directly applied to the lambda, meaning a multiplier value greater

Setting

Definition

than 1 adds distortion and reduces quality, and a multiplier value between 0 and 1 reduces distortion and adds quality. The result is rounded to the nearest integer for use. This can not disable RDO (the result will be at least 1).

`bForceAllBC23ToBC7`

When enabled, any time a BC2 or BC3 (i.e. DXT3 and DXT5) format is requested, Oodle will compress a BC7 texture instead. BC7 is usually better quality, but depending on your minimum specifications might not be available.

`bForceRDO0ff`

When enabled, RDO will be disabled at all times, independent of any texture-specific RDO setting.

`bDebugColor`

When enabled, textures will be compressed as a solid color representing their encoded format for visual identification.

Format	Color
BC1	Red (0xff0000)
BC2	Dark Green (0x008000)
BC3	Green (0x00ff00)
BC4	Dark Yellow (0x808000)
BC5	Yellow (0xffff00)

Setting


Definition

Format	Color
BC6	Purple (R = .5f, G = 0.0f, B = .8f)
BC7 (Opaque)	Dark Blue (0x8080ff)
BC7 (Transparent)	Blue (0x0000ff)

Configuring Texture LOD Groups

The LOD Group parameter representing RDO lambda is called **Lossy Compression Amount**. This parameter is defined for LOD groups in the `DefaultDeviceProfiles.ini` file.

```
1
2 TextureLODGroups=
  (Group=TEXTUREGROUP_WorldNormalMap,MinLODSize=1,MaxLODSize=8192,LODBias=0,MinMagFilter=aniso,MipFilter=
3
```

 Copy full snippet

Lossy Compression Amount can take the following values:

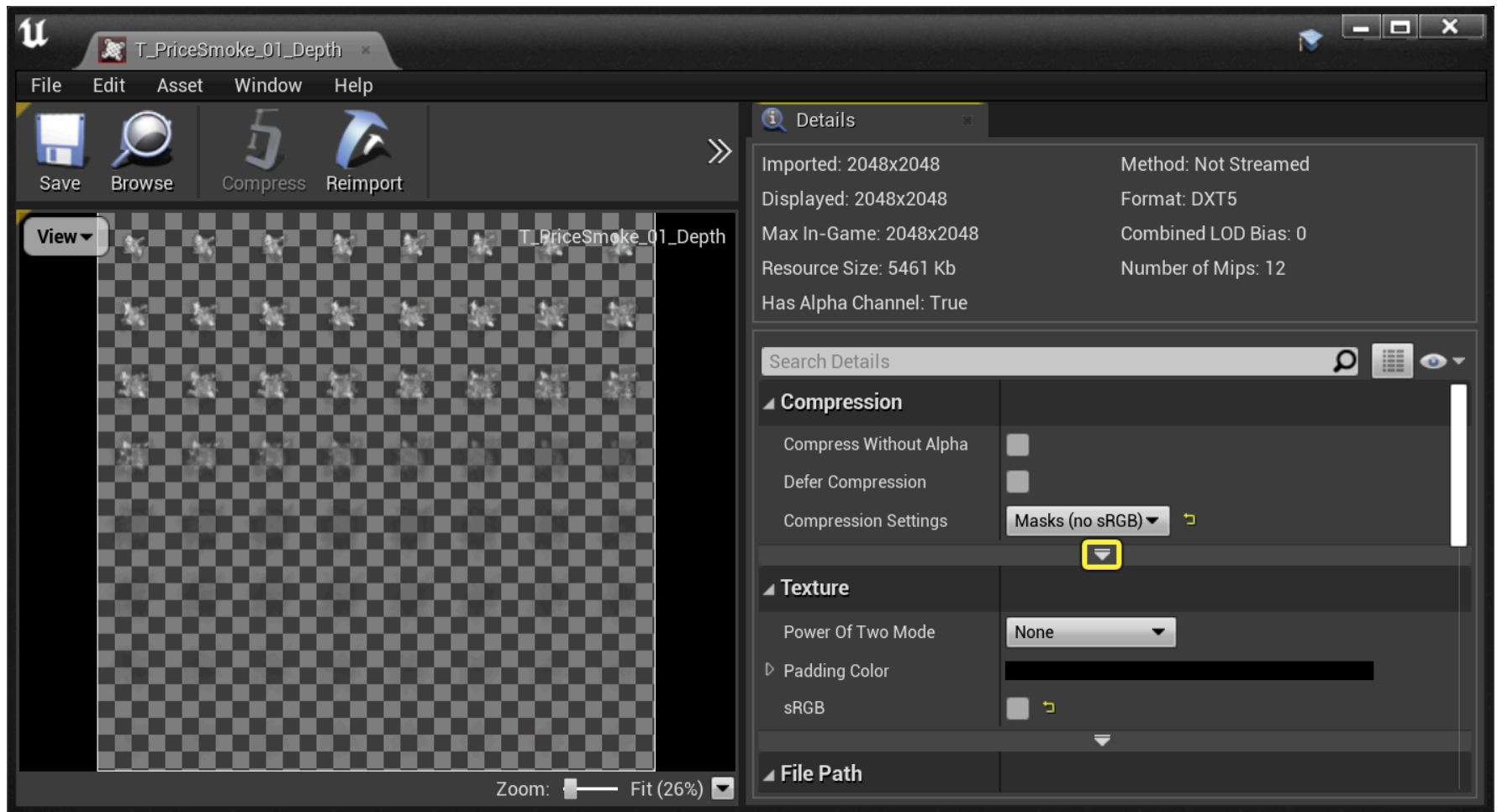
Value	Description
TLCA_Default	Inherit. If set on a texture, inherit from the LOD group. If set on an LOD group, inherit from <code>DefaultRDOLambda</code> .
TLCA_None	Disable RDO (0)
TLCA_Lowest	5 - Best quality, largest file size.
TLCA_Low	15
TLCA_Medium	30
TLCA_High	40
TLCA_Highest	60 - Worst quality, smallest file size.

Configuring Individual Textures

RDO Lambda can also be set for individual textures using the Lossy Compression Amount parameter, and takes the same values shown above.

To set the parameter for a single texture:

1. Double-click the texture you want to set RDO Lambda for in the **Content Browser** to open it in the **Texture Editor** window.
2. In the **Details** panel, expand the **Compression** section and click the arrow icon to show the **Advanced** options.



3. Use the dropdown menu next to the **Lossy Compression Amount** parameter to select the desired value.

