

# V-Ray Bake

This page provides information on the V-Ray light bake process with V-Ray for Unreal.

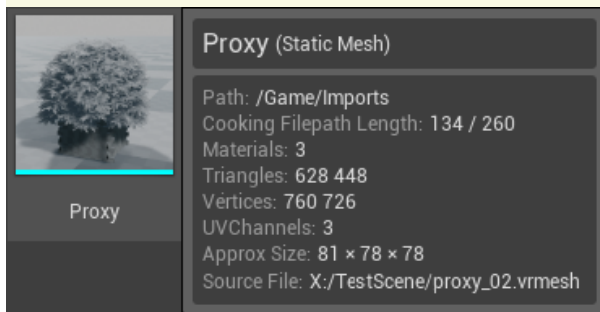
## Overview

V-Ray Light Bake or just V-Ray Bake is the process of creating lightmaps with complex light interactions and diffuse interreflection using the V-Ray render engine and it's GI and Lighting calculations.

The light bake quality depends on the following:

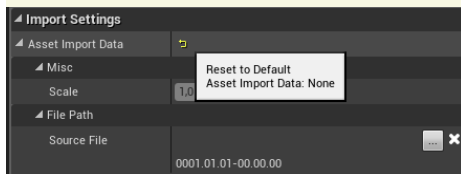
- the lightmap resolution of the meshes;
- the lightmap UVs quality, better unwrapped UVs better bake quality;
- the V-Ray settings in the settings tab.

Note that actors with vmeshes assigned will not have their lightmaps baked but will still contribute to the scene for shadows and GI



However, it is possible to import a proxy and bake it with a light map, provided the mesh has correct UV mapping.

Select the proxy and go to **Static Mesh > Import Settings** rollout. Press the **Reset to Default** button, which converts the proxy to a static mesh. The static mesh can now be baked.

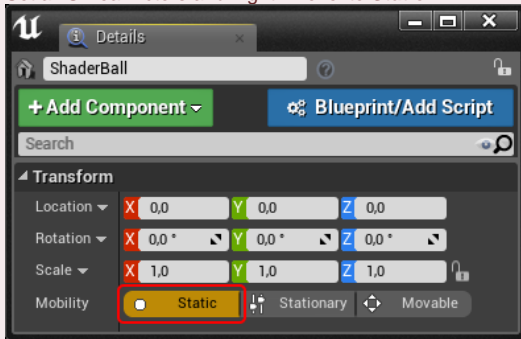


Please note that by doing this you will lose the benefits of memory optimization when using vrmesh. Using static meshes instead of vmeshes **will require more RAM resources**.

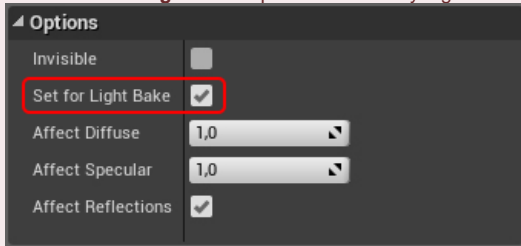
Please check the following before V-Ray Bake:

- Read the System Requirements
- Overlapping/wrapping and/or large amount of polygons in static meshes may result in longer bake times and/or crashes

- Set all Unreal Actors and Light in level to **Static**



- Enable **Set for Light Bake** option for all V-Ray Lights



Viewport  
VFB

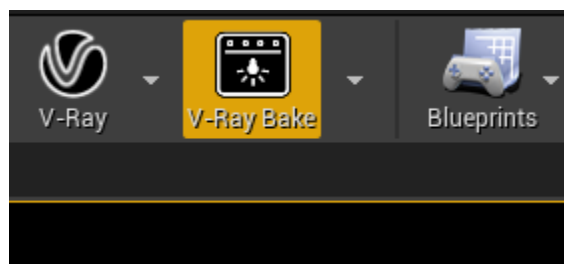




## UI Paths

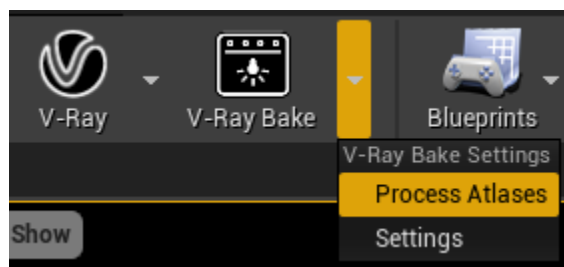
You can initiate a V-Ray Bake from the toolbar by clicking on the **V-Ray Bake** button.

||Toolbar|| > **V-Ray Bake**



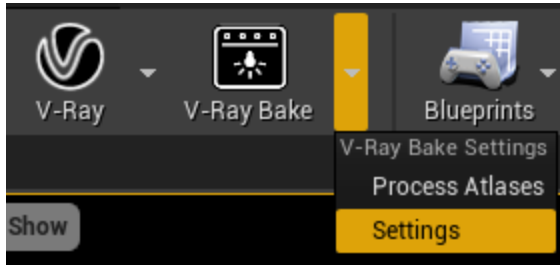
The Process Atlases functionality is accessible through the drop down menu on the right side of the **V-Ray Bake** button.

||Toolbar|| > **V-Ray Bake Settings dropdown** > **Process Atlases**



The V-Ray Bake Settings tab is accessible through the drop down menu on the right side of the **V-Ray Bake** button.

||Toolbar|| > **V-Ray Bake Settings**  
dropdown > Settings



## Bake Settings

Below are mentioned all the settings and parameters that modify the quality of the V-Ray Bake from the **V-Ray Bake Settings** tab and the **V-Ray Settings** tab.

### V-Ray Bake Settings Tab

#### Bake Settings

**Light bake quality** – A slider for a set of quality presets.

**0 - Preview** - Progressive Image Sampler; minSubdivs - 8; maxSubdivs - 32; Sample Limit - 100; Noise threshold - 0.3; Light Cache subdivs - 300; Retrace - 2; Denoiser -Strong denoising with strength 2 and radius 15

**1 - Medium** - Bucket Image Sampler; minSubdivs - 8; maxSubdivs - 32; Sample Limit - 1000; Noise threshold - 0.1; Light Cache subdivs - 1200; Retrace - 4; Denoiser - Default denoising with strength 1 and radius 10

**2 - High** - Bucket Image Sampler; minSubdivs - 8; maxSubdivs - 32; Sample Limit - 2000; Noise threshold - 0.005; Light Cache subdivs - 3000; Retrace - 8; Denoiser - Default denoising with strength 1 and radius 10

**+ NEW - 3 - Custom** - Uses the settings from **V-Ray Setting** tab.

**Lightmap Atlas resolution** – Sets the preferred atlas resolution.

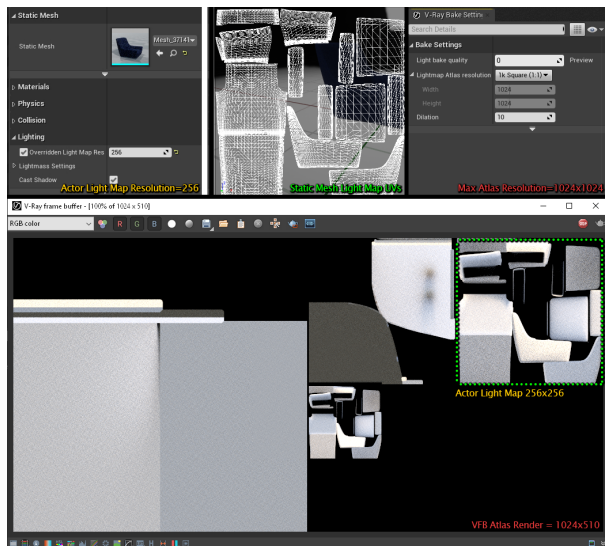
Note that static meshes that have larger Light Map Res will automatically resize the atlas.

**Presets: 1k Square (1:1); 2k Square (1:1); 4k Square (1:1); Custom**

Large Atlas Resolution values speed up light baking but increase memory consumption.

The number of light map atlases depends on:

- Actor Light Map resolution
- Atlas resolution
- Actor count in the level



**Dilation** – The number of pixels to expand around geometry UVs.

**Export VRScene** – Enables the export of a *.vrscene* file when the V-Ray Bake button is clicked.

**Export VRScene Path** – Directory where the *.vrscene* file is saved with the file name *bake#.vrscene*. Default directory is *<Project\_Path>*

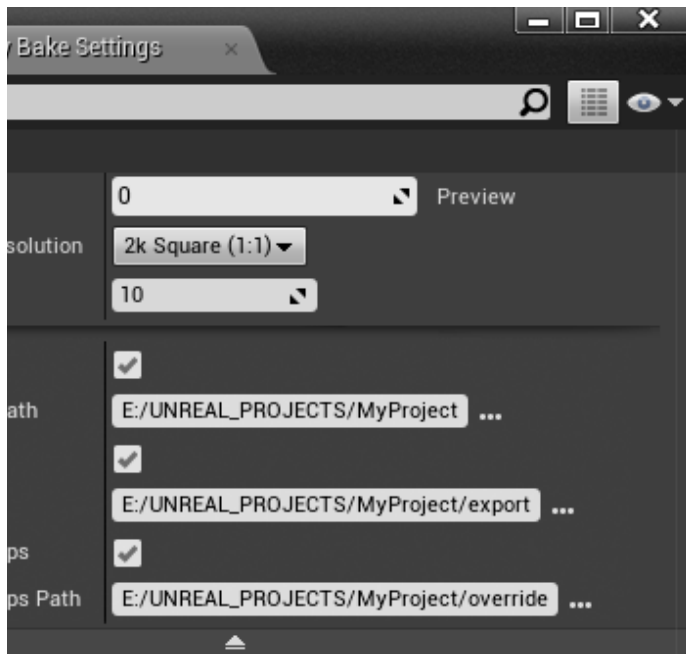
**Export EXR** – Enables the export of a *.exr* file for each Static mesh Actor in the level when the V-Ray Bake button is clicked.

**Export EXR Path** – Directory where the *.exr* files are saved with the file name *<StaticMeshActor\_Name>.exr*. Default directory is *<Project\_Path>/export*

**Override Light Maps** – Enables the loading of *.exr* files for Static Mesh Actors in the level when the V-Ray Bake button is clicked.

**Override Light Maps Path** – Directory from where the *.exr* files are loaded when the V-Ray Bake button is clicked. Default directory is *<Project\_Path>/override*.

Only Static Meshes that do not have appropriate *.exr* files in the directory specified in the **Override Light Maps Path** will be included in the baking process.



If you have **Export EXR** option enabled before the bake afterwards you can rebake only selected actors without the need to bake the whole level. After the initial bake you can copy the **.exr** files from the directory specified in **Export EXR Path** to the directory specified in **Override Light Maps Path**. After you made your modifications to the actors (e.g. increase Lightmap resolution) enable **Override Light Maps** option and then delete the appropriate **.exr** files (<StaticMeshActor\_Name>.exr) from the directory specified in the **Override Light Maps Path**. When you click the V-Ray Bake button the lightmaps for the actors will be loaded from the **Override Light Maps Path** folder and only the Actors that do not have according **.exr** files will be baked.

You can use the **Export EXR** and **Override Light Maps** options to have a resumable lightbake. When **Export EXR** is enabled after each successful atlas bake **.exrs** are saved in **Export EXR Path** directory. In cases of a crash or something else you will not lose your progress. Just copy the **.exr** files from the directory specified in **Export EXR Path** to the directory specified in **Override Light Maps Path** and enable the **Override Light Maps** option. When you click the V-Ray Bake button the baked **.exr** lightmaps will be loaded from the **Override Light Maps Path** folder and the bake will continue from when it has stopped.

## V-Ray Settings Tab

### Camera Settings

**Auto Correction Mode** – Specifies which camera to use the automatic exposure and white balance setting.

**All cameras**  
**Cameras without exposure only**

**Auto Exposure** – Automatically determines an appropriate exposure value for the render. It requires Light Cache to be set as a GI engine and **Interactive** turned off.

**Auto White Balance** – Automatically determines a suitable white balance value for the image. It requires Light Cache to be set as a GI engine and **Interactive** turned off.

Auto Exposure and Auto White Balance work only when using Light Cache as GI solution.

### Render Settings

**Time Limit (Minutes)** – Specifies the maximum time (in minutes) for refining the image.

### Global Illumination

**GI Engine** – Specifies the type of secondary engine. Note that Brute force is always used as a primary engine.

**Brute Force**

**GI Depth** – The number of bounces for indirect illumination.

### Light Cache

**Environment** – Specifies which [V-RaySettingsEnvironment](#) blueprint will be used at render time for environment overrides. Blueprint must be picked from the level and assigned using either the drop-down list or the picker.

### Lights

**Shadow Bias** – A global parameter for lights. This value moves the shadow toward or away from the shadow-casting object (or objects). Higher values move the shadow toward the object(s) while lower values move it away. If this value is too extreme, shadows can "leak" through places they shouldn't or "detach" from an object.

### Materials

**Fallback Material** – Specifies what material to be used at render time for unsupported materials. If the slot is empty actors that have unsupported materials assigned will be rendered transparent.

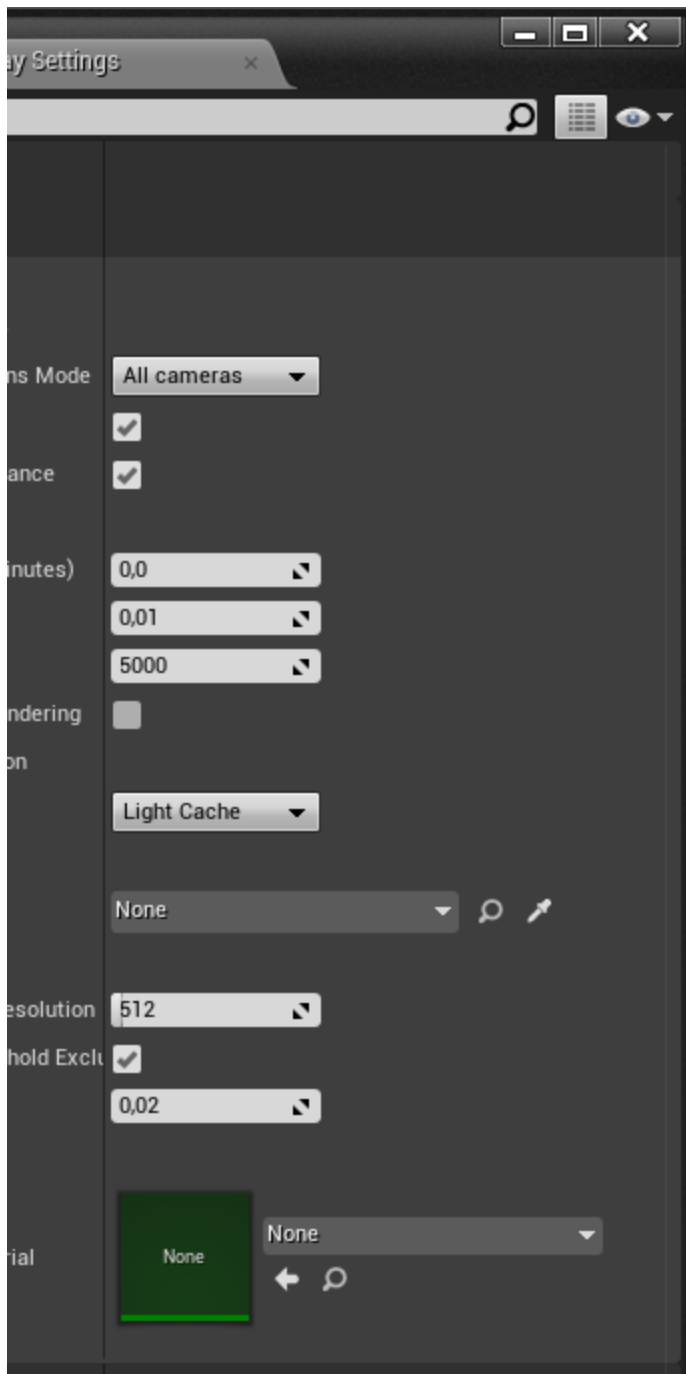
### Distributed Rendering

**On** – Enables or disables distributed rendering for V-Ray for Unreal. Use the **Render Servers** list to specify the machines that will be used for distributed rendering.

**Use Local Host** – When enabled, the client machine (the one from which the user initiates the DR render) will also take part in the rendering calculations. If disabled, the client machine only organizes the DR process, sends rendering tasks to the render servers and displays the final result.

**Render Servers** – Specifies the render servers. Click the **+** button to manually add a server by entering its IP address or network name followed by its port number (default is 20207) (e.g. 127.0.0.1:20207)

**Maximum cached asset size** (in GB) – This option specifies the maximum assets cache size limit in GB on the render machines. When the size of the assets on a render server exceeds the specified amount, least recently used assets are automatically deleted at the end of each render until the folder size is less than the limit.



## Process Atlases

Process Atlases loads Light bake atlas .exr files and extracts the lightmaps for actor that are in that atlas. Those .exrs are a result of rendered exported .vrscene files from the **V-Ray Scene Export Options** dialogue using the **Light Bake** option.

When you click on the Process Atlases button in the **V-Ray Bake Settings** drop down a browse window will appear where you can select the atlas .exrs . In order for the Process Atlases to work you must first enable the **Export EXR** option in the **V-Ray Bake Settings** tab and set the **Export EXR Path** where the lightmaps per actor will be extracted.

You can use the **Process Atlases** workflow to generate atlas .vrscenes and render them when Unreal is closed to save on resources that it needs or on a renderfarm separately. After actor lightmaps .exrs are extracted from the atlas .exrs use the **Override Light Maps** options with correct set **Override Light Maps Path** pointing to the extracted actor lightmaps to load the actor light maps. Click the **V-Ray Bake** button and the baked lightmaps will be loaded from the **Override Light Maps Path** folder and applied to according actors in the level.



## Bake Progress

---

The V-Ray Bake button has five states depending on the process that is currently running:



### Idle/Neutral

V-Ray is in idle mode, not processing or rendering the scene.



### Preparing

V-Ray is preparing the scene for baking. Meshes, textures, materials and lights are cached into memory.



### Rendering

V-Ray is baking the scene. Click on the button to stop rendering at any time.



### Busy

V-Ray is unloading resources, exits rendering and processes lightmaps.



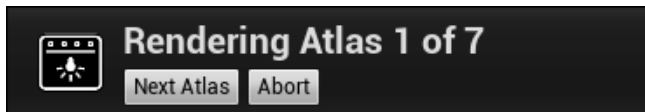
### Finished

V-Ray has finished  
baking all atlases and  
lightmaps.

When light baking a progress bar appears in the lower right corner of the viewport. It shows which atlas is currently rendering and how much atlases are there in total.

**Next Atlas** – Stops the current atlas rendering, runs the denoiser and continues to the next atlas. This functionality is **only** available when in **Preview** preset.

**Abort** – Stops the current atlas rendering and closes the frame buffer without applying the rendered lightmaps.



## Usage Examples

---

In this short example, a `.vrscene` file is loaded. Before we bake the lighting we first make sure that all lights in the scene are set to Static. Next we click on the **Bake** button in the V-Ray drop-down. The VFB pop up and the lightmaps begin to render. When it is done they are encoded and imported into Unreal. Finally, we start the IPR rendering and start walking around to see the parity between the lightbake and the actual render.