

H-Trace

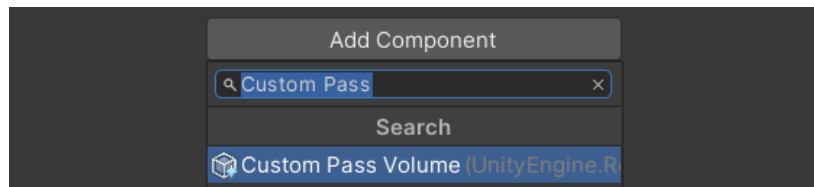
Quick-start manual

This is a short manual that covers basic things and helps you to get started with H-Trace. If you have any questions, bug reports or suggestions - feel free to reach out to us using [Unity Forum thread](#) or [Discord](#).

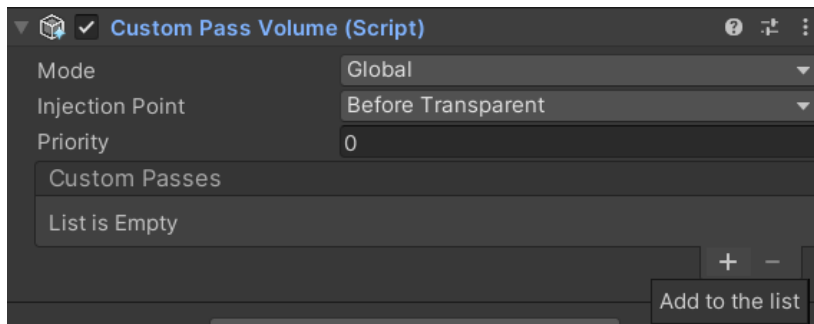
First Steps

When the package is installed, navigate to the folder called “**Sample Scene (Cornell Box)**” that you can find inside the main H-Trace directory. This folders contains an example scene with a Cornell Box environment and a preconfigured custom pass volume. Taking a look at this scene is the fastest and the easiest way to see how the system is set up.

To set up H-Trace in a new scene do the following steps:



1. Create an empty game object (or use an existing one) and add the **Custom Pass Volume** component to it.



2. Use the **Add to the list** button to open a list of available custom passes and select “**H Trace pass**”.



3. If you have any **reflection probes** in your scene (especially the real-time ones), make sure to disable the **Custom Pass** option in the **Frame Settings Overrides** menu

After completing these steps make sure that the **Injection Point** parameter of the custom pass volume is set to either “**Before Transparent**” or “**Before Post Process**”. It’s not recommended to use other injection points, unless your HDRP package is patched.

Patching HDRP Package

H-Trace doesn't support all rendering features out of the box. When first installed and configured according to the steps above, the following features **will not** be available:

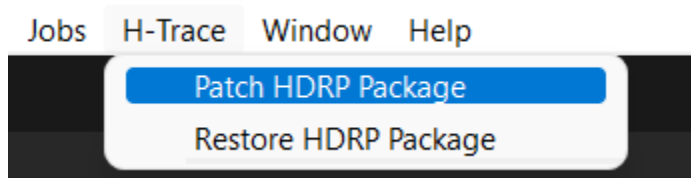
- Forward rendering & Forward materials support;
- Interaction with Unity's SSR (GI visibility in screen-space reflections);
- Ground Truth Specular Occlusion.

HDRP package must be patched to enable these features. The patching process is automated and doesn't affect any files outside of the scope of HDRP package, but if you feel unsure about this operation - it's recommended to first try it on a separate project before applying it to your main working directory.

Besides enabling the features listed above, **patching has following consequences**:

1. Your HDRP package will be moved to the local directory (unless it's already there).
 2. The Custom Pass injection point called **"Before Pre Refraction"** will be internally moved to another place inside the Render Graph. If you have other Custom Pass effects executed on this point - they are likely to break.
-

The patching process can be launched by pressing the **"Patch HDRP Package"** button in the upper H-Trace menu:

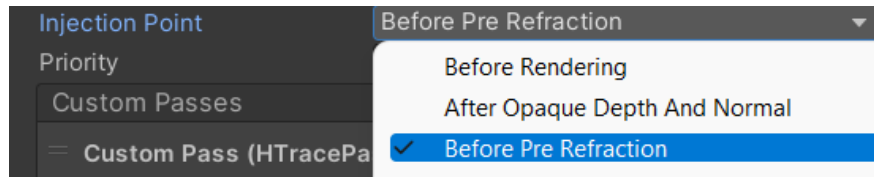


During the patching process you're likely to see a number of console errors and warnings. The rendering in the main view may also become broken. This is caused by Unity temporarily losing its connection with the HDRP package and will be automatically resolved during the next editor loading, so there's no reason to worry about it. When the patching process is complete you will be asked to reload the Unity Editor.

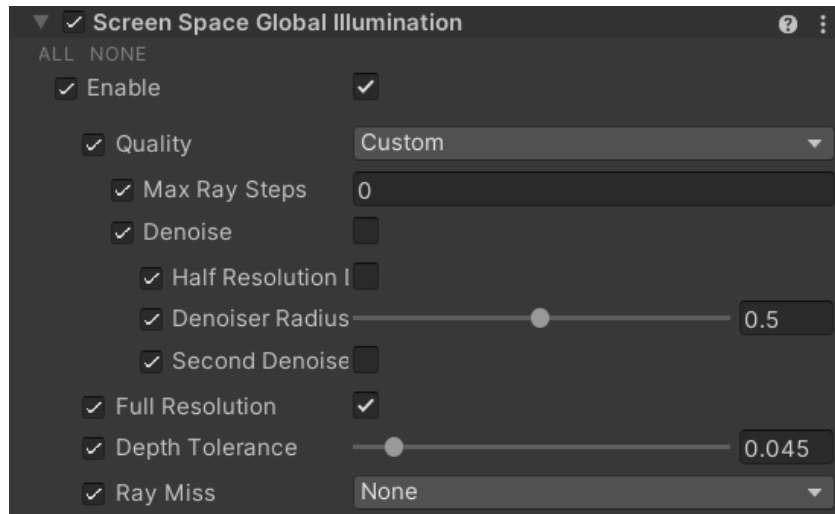
After reloading the editor you may still encounter certain errors coming from some of H-Trace shaders. To resolve these errors, select all shaders in the **"Assets\Resources\H-Trace"** folder and in the **"Assets\H-Trace\Shaders"** folder, right-click on them and press the **Reimport** button in the context menu.

After the Patch is Applied

After the patching process is completed, you may start using additional features offered by H-Trace. To do so, follow these steps:



1. Change the **Injection Point** of the H-Trace custom pass to the **"Before Pre Refraction"**



2. Add the **"Screen Space Global Illumination"** override to your volume and **configure it** as shown in this screenshot

When patched and executed on the **Before Pre Refraction** injection point, H-Trace will act through Unity's native SSGI system. More specifically:

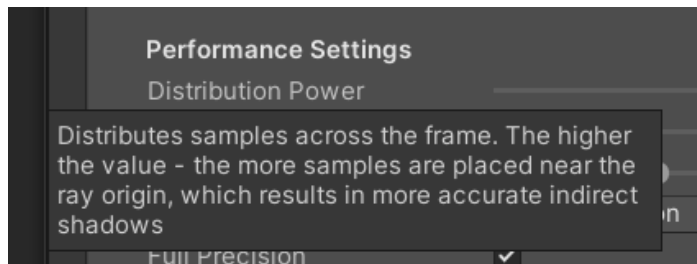
1. You have to keep SSGI override **enabled** when using H-Trace in this mode.
2. SSGI shaders are internally overridden by H-Trace and therefore you **will not** be paying double performance cost for both GI solutions running simultaneously.
3. **"Max Ray Steps"**, **"Depth Tolerance"** and **"Ray Miss"** parameters are internally overridden and their values will have zero impact when H-Trace is enabled.
4. It's recommended to keep the **"Full Resolution"** parameter enabled. If you disable it, the output of H-Trace will indeed be downscaled, but with no additional performance boost. If you want to enable a true resolution downscale with the actual performance gain - do so in the H-Trace settings (in the custom pass volume).
5. All SSGI denoising parameters **are not** overridden. If you enable the **"Denoise"** checkbox in the SSGI override - Unity will apply its own denoising on top of the H-Trace output. Don't forget to disable the H-Trace denoising tab in this case to avoid double denoising which is not recommended (but not forbidden) for performance reasons.
6. You can switch between Unity's SSGI and H-Trace at any time: SSGI is automatically restored when H-Trace is disabled or set to any other injection point than **"Before Pre Refraction"**. When H-Trace is enabled and set to the required injection point, SSGI is automatically overridden again.

Final Notes & Useful Tips

If you want to revert the changes applied to the HDRP package during the patching process - use the **“Restore HDRP Package”** button in the H-Trace menu. If you need to update the HDRP package, follow these steps:

1. Remove the package from the local directory
2. Download the new version from the Package Manager as usual
3. Apply the patch again (if needed)

All parameters of the H-Trace custom pass have tooltips with description. Hover your mouse over any parameter to find out what it is used for:

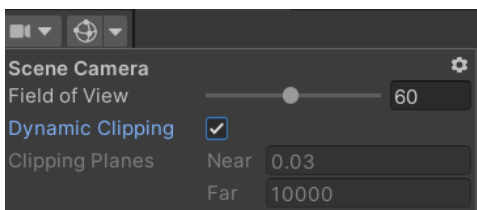


A tooltip example for the “Distribution Power” parameter

Use the **“Debug Mode”** dropdown to inspect different parts of H-Trace, such as AO, SO, Normals and GI. The debug menu is available only on the “Before Transparent” injection point or higher. If your pass is running on the “Before Pre Refraction” point - temporarily switch it to one of the higher points for debugging.

When using the **“Accurate” Thickness Mode** you may see an increase in the polycount. This is due to an additional rendering of the scene into a custom depth buffer. The cost of this rendering is expected to be really low, because all objects use a single extremely lightweight (essentially empty) shader during this operation. However, if you want to exclude some object(s) from this rendering - you can do this on a per-layer basis using the **Thickness Layer** dropdown. Note that objects with no real thickness (such as planes) will fall back to the infinite thickness (a.k.a “Disabled” Thickness Mode). It’s recommended to exclude such objects as well to avoid over-occlusion and over-shadowing.

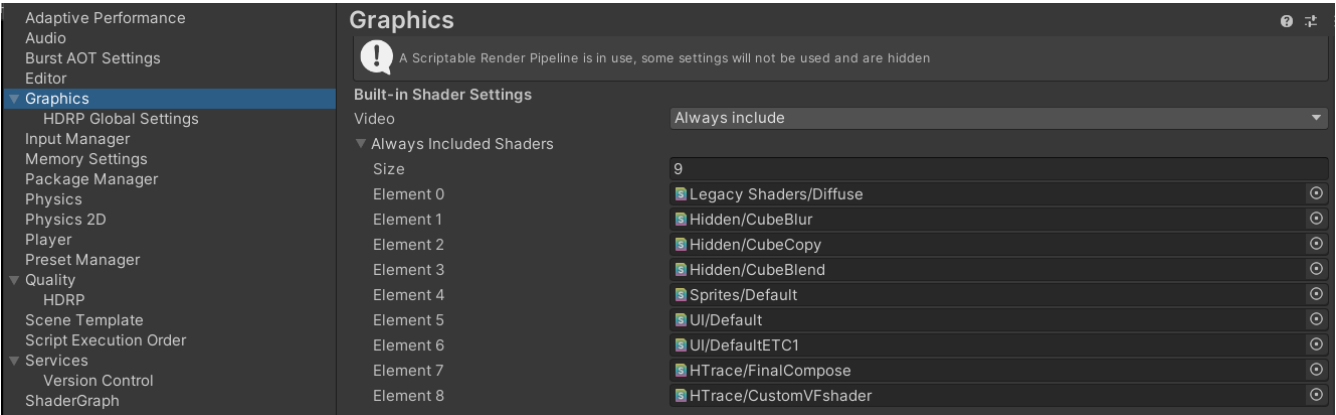
Specular Occlusion is more accurate when the **“Slice Count”** value is set to **2 or above**. Consider using Slice Count = 2 and Resolution Scale = Checkerboard to achieve the **same performance** with **better visual results** than with Slice Count = 1 and Resolution Scale = None.



If the GI effect in the **Scene** tab looks different than in the **Game** tab - it’s recommended to check the **clipping** settings in the **Scene Camera** menu (don’t confuse it with your main camera object). **Dynamic Clipping** is the preferable mode.

Build Notes

1. Read the official [Unity guide](#) about custom passes to prevent and troubleshoot issues.
2. It's recommended to include H-Trace shaders from the **"Assets\H-Trace\Shaders"** folder into the **Always Included Shaders** list as shown in this screenshot:



H-Trace shaders are listed as **Element 7** and **Element 8**.