CTI Runtime Components HDRP HLSL

About this documentation

In case you want to use CTI trees along with the HDRP you have to assign the *CTI HDRP* shaders and use the *CTI_SRP_CustomWind* script. Both shaders and script are slightly different from the CTI Runtime Components for the built in RP.

Table of Content

Limitations

- Shaders need Unity 2019.4. and HDRP 7.3. or above. Unity 2020.1.0f1 and HDRP 8.2 as well as HDRP 9.0.0 preview.71 have been tested as well. Other versions have not been tested.
- Only basic LOD trees are supported. Tessellation is not supported..
- The shaders only accept wind from script.
- You can not author billboard textures using HDRP nor can you use the debug shader, so authoring should take place using the built in SRP.

Changes

- The leaf shader uses the **built in transmission lighting.** So you have to add/edit a diffusion profile.
- **Specular color** has been dropped. This is driven by the *Index of Refraction* in the diffusion profile..
- The **leaf shader** uses a **regular normal** or bump map. Lighting uses the built in transmission feature.

- The shaders only accept **wind from script**. → *The Tree component is not needed at all and should be removed.*
- The CTI LOD HDRP shaders need a slightly different input for the **wind from script**. → *You have to use the CTI_SRP_CustomWind script instead of the old one.*
- **Fade out Wind** has been dropped.
- **Fade out Translucency** has been dropped.
- **Tumbling** and **Turbulence** have slightly been reworked and optimized. **Leaf Noise** has been added. → *You may have to adjust their settings.*
- **Wind multipliers** for primary and secondary bending as well as edge fluttering have been added. → *Now you can tweak the bending without editing the tree. Make sure multipliers in the bark material match those in the leaf material.*

CTI HDRP HLSL LOD Shaders

The shaders were authored using Shader Graph but were converted to HLSL in order to work around shader graph's LOD dithering bug during the transitions.

CTI HDRP/Bark HLSL

Shader Inputs

Surface Options

Surface Type Opaque

Rendering Pass *Should be Default.*

Alpha Clipping *Should be unchecked.*

Double-Sided *Should be unchecked.*

Exposed Properties

~~**Color Variation (RGB) Strength (A)** Color variation in RGB. Alpha contains the strength.~~

Albedo (RGB) Smoothness (A) Diffuse texture which contains **smoothness** in the alpha channel.

Normal Map (GA) Occlusion (B) contains the combined normal and occlusion map. *Red color channel should be black.*

Normal Strength Lets you adjust the strength of the normal.

Smoothness Multiplier for the smoothness as sampled from the *Albedo (RGB) Smoothness (A)* map.

Wind Multipliers

X Multiplier for the Primary Strength. *Must match the value in the leaf material.*

Y Multiplier for the Secondary Strength. *Must match the value in the leaf material.*

Z Multiplier for Edge Flutter. *Does not matter here.*

CTI HDRP/Leaves HLSL

Shader inputs

Surface Options

Surface Type Opaque

Double-Sided Please check if your leaf geometry is only single sided (recommended)

Normal Mode Should be set to *Mirror*.

Exposed Properties

~~**Color Variation (RGB) Strength (A)** Color variation in RGB. Alpha contains strength.~~

Albedo (RGB) Alpha (A) Diffuse texture in RGB, opacity or alpha in A.

Alpha Cutoff Cutoff value.

Normal Map The Normal Map.

Normal Scale Scale of the normal.

AO (G) Translucency (B) Smoothness (A) contains the combined ambient occlusion, normal, translucency and smoothness map. ***Please note:*** *In HDRP translucency is an inverted thickness map. Thickness may have to be adjusted according to your diffusion profile.*

Smoothness Min / Smoothness Max Lets you tweak the smoothness as sampled from the *Normal (GA) Smoothness (B) Trans (R)* map.

Thickness Lets you remap the thickness as sampled from the *AO (G) Translucency (B) Smoothness (A)* texture. Unlike smoothness Min / Max this is just a multiplier.

Wind Multipliers

X Multiplier for the Primary Strength. *Must match the value in the bark material.*

Y Multiplier for the Secondary Strength. *Must match the value in the bark material.*

Z Multiplier for Edge Flutter.

Tumble Strength defines the strength of the tumbling animation.

Tumble Frequency lets you adjust the frequency of the tumbling.

Leaf Turbulence lets you adjust the strength of the turbulence.

Leaf Noise lets you adjust the strength of the edge flutter (stored in vertex color green) affecting the leaf turbulence. Using edge flutter influence values above 0.0 will most likely add some distortion to the leaf meshes – which in fact looks really nice.

CTI HDPR/Billboard HLSL

Shader inputs

Surface Options

Surface Type Opaque

Double-Sided Must be checked! HDRP renders back faces?!

Normal Mode Must be set to ***Mirror***! According to back faces being rendered here...

Exposed Properties

~~**Color Variation (RGB) Strength (A)** Color variation in RGB. Alpha contains strength.~~

Albedo (RGB) Alpha (A) This slot should contain the created albedo (RGB) and the alpha (A) texture atlas.

Alpha Cutoff Cut out value.

Normal (AG) Translucency (R) Smoothness (B) This slot should contain the created texture atlas.

Normal Scale Scale of the normal.

Smoothness Multiplier for the smoothness as sampled from the *Normal (AG) Translucency (R) Smoothness (B)* map.

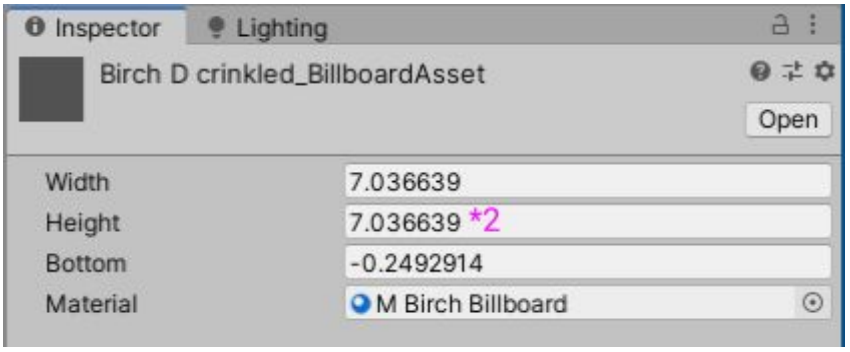
Wind Strength As Billboards do not have any baked wind information you may use this parameter to make the bending of the billboard better match the bending of the mesh tree.

Wind Power Power value which drives the wind strength along the y axis. Should match the power value used on importing the tree. Default is 1.5.

Thickness Remap Lets you remap the thickness as sampled from the translucency channel. Nevertheless the bark will never get fully opaque when using the built in Foliage diffusion profile… *Consider creating a diffusion profile just for billboards. A diffusion profile which contains the range from 0 - 8 should be fine to give you fully translucent and opaque parts while keeping thickness in a nice 0-255 range.*

Vertical Scale Actually CTI contains a bug: Billboard assets only have half the height they need (so they are most likely as high as they are wide) and for some reason i can't remember i never fixed this on the asset but simply added a fixed multiplier of 2 in the shader. While this is visually correct it might cause issues if the billboard leaves the screen at the lower border as it will be culled too early.
So with this factor exposed you can fix the billboards:

- Edit die billboard asset and add "*2" at the end of its height value. Unity will calculate the proper final height which in this case is 14.07328.



- Then edit the billboard material and set *Vertical Scale* to 1.

LOD Settings

In case you want to use LOD groups and smooth LOD fading you have to set the *Fade Mode* to *SpeedTree*. *Fade Mode = Cross Fading* is not supported as HDRP can not handle this properly.

HDRP seems to ignore(?) “Animate Cross-fading” or at least adds a very short animated cross fade distance. So i recommend to check this off and manually tweak the cross fade distance between the last mesh LOD and the billboard by editing the “Fade Transition Width”.

Please note: You have to absolutely make sure that the LOD level occupied by the billboard uses a “Fade Transition Width” close to 0. Otherwise billboards will fade out gradually and not be batched - which is super bad for performance.

Only the last mesh based LOD level should use a custom “Fade Transition Width”.