



EASY SKY

1.0.0

Documentation

Thank you for your purchase!

ASSIST
Innovative Minds



OVERVIEW & INSTALLATION

Render Pipeline Setup	3
Installation	4
Global controls	5
Global sky system	6
Weather area	9
Weather presets	10
Unity Store Ready Integrations	16

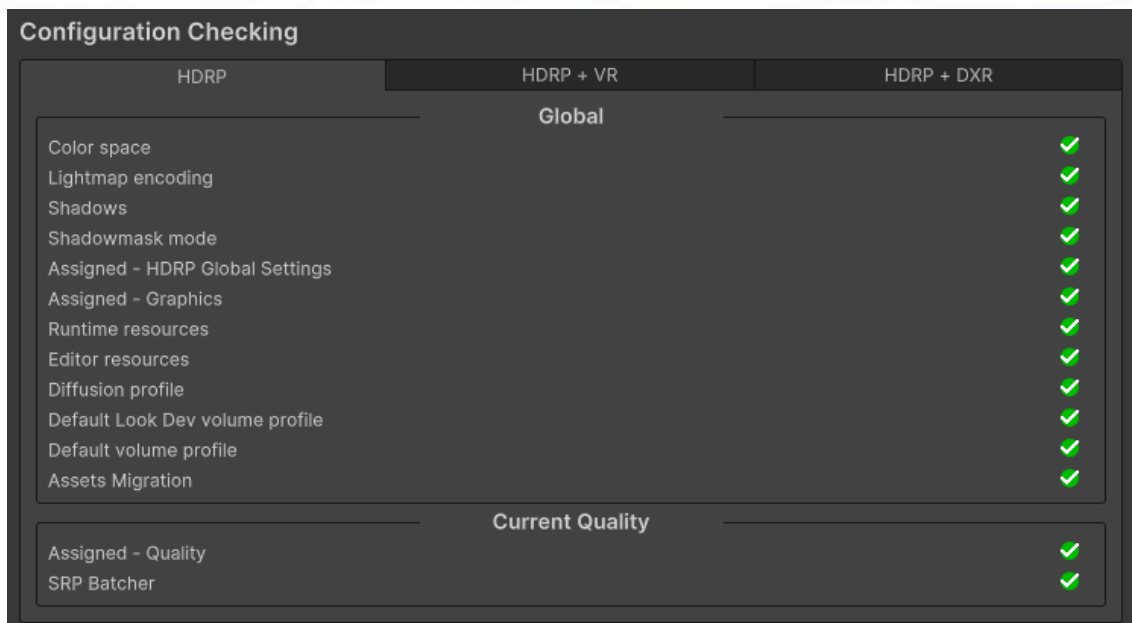
SCRIPTING API

Easy Sky Weather Manager	17
Full-screen and Custom Pass Effects	20
Natural Phenomena Particle system	25
FAQs	26

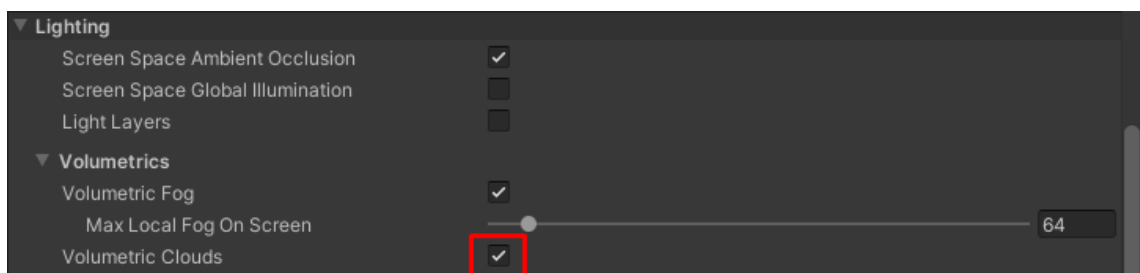
Overview & Installation

Render Pipeline Setup

Before importing Easy Sky plugin, you should ensure that HDRP setup is complete. From Unity top-bar, select Window -> Rendering -> HDRP Setup



Also, you should ensure that Volumetric Clouds is enabled. You can access it from Unity top-bar, Edit -> Project Settings -> Quality -> HDRP. On the Lighting section, check "Volumetric Clouds"

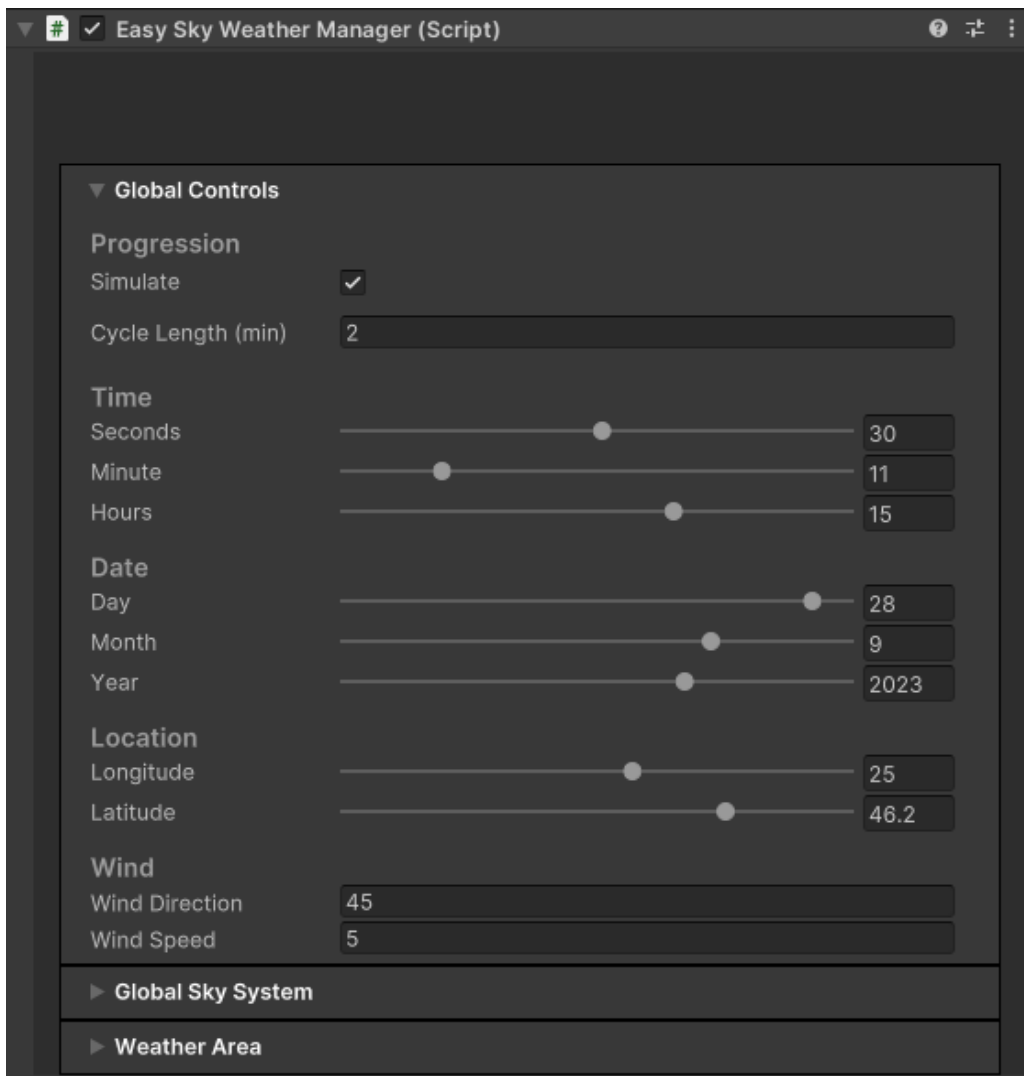


Installation

Import the Easy Sky package and follow these steps to get started:

—

On the Unity top bar, you will find a new tab “EasySky”, so click on it and then select “Weather Manager”. A new GameObject will be created in your scene with the “EasySkyWeatherManager” component.



Global controls

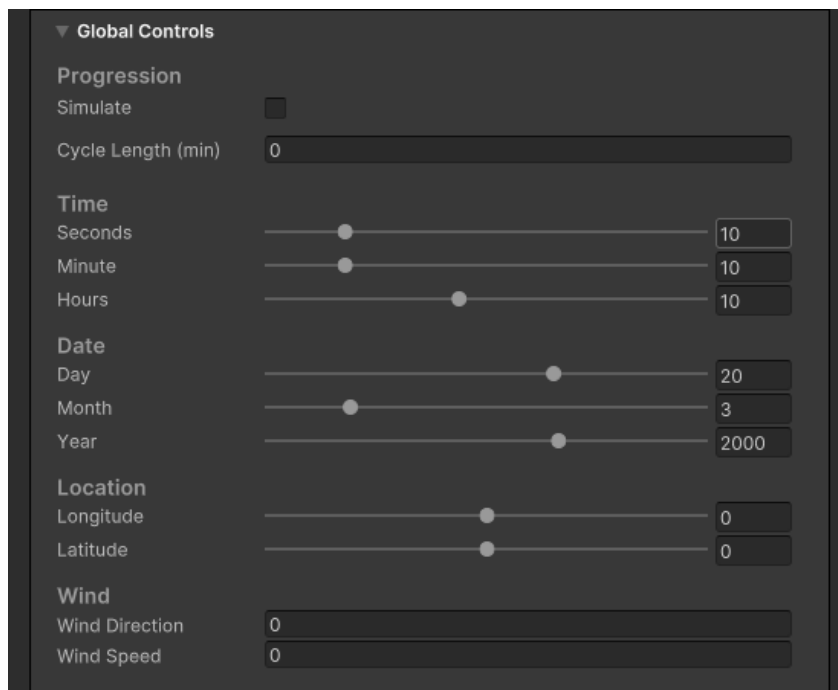
All the settings in this section are applied across the entire system.

The behaviour of the "Simulate" toggle is contingent upon whether the time is set as static or dynamic. To modify the pace of time, you can manipulate the "Cycle length," which determines the duration of a day in minutes.

The manipulation of time and date is facilitated through the utilization of the six sliders and input fields at your disposal.

Altering the location necessitates the utilization of both the "Latitude" and "Longitude" sliders, in conjunction with the corresponding input fields.

For wind adjustment, you can modify both the Direction (in degrees, following the same measurement as Unity) and the speed, which is measured in units per second.

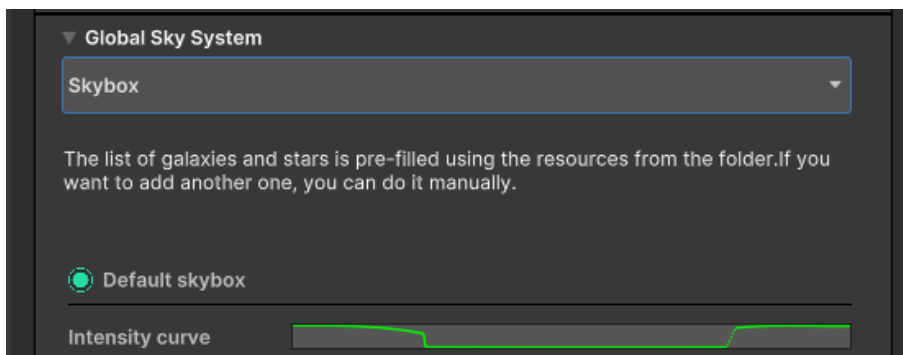


Global sky system

Within this section, you will discover all the settings associated with the skybox.

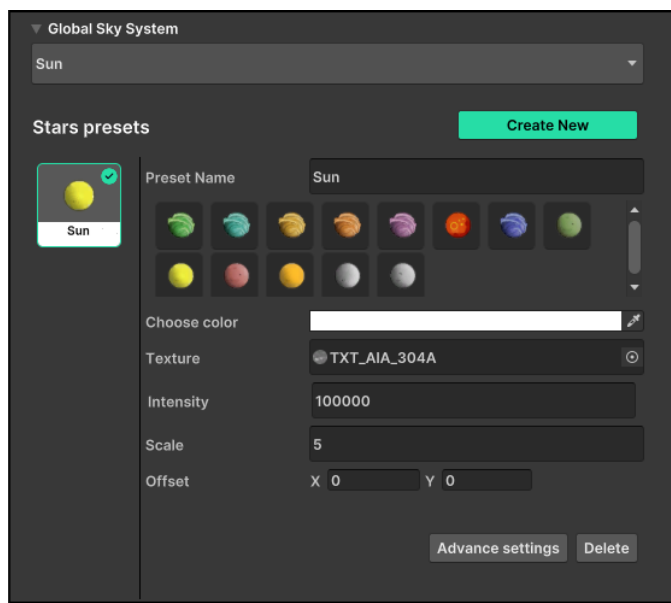
Skybox

Here, you have the option to select from various sky cubemaps while also configuring a custom animation curve to manage the luminosity intensity. To incorporate additional skyboxes, simply include the cubemap within the designated Resources/Skybox folder.



Sun

Here you can create and edit presets for nearby stars



Star settings

Color: the color of the light emitted by the star.

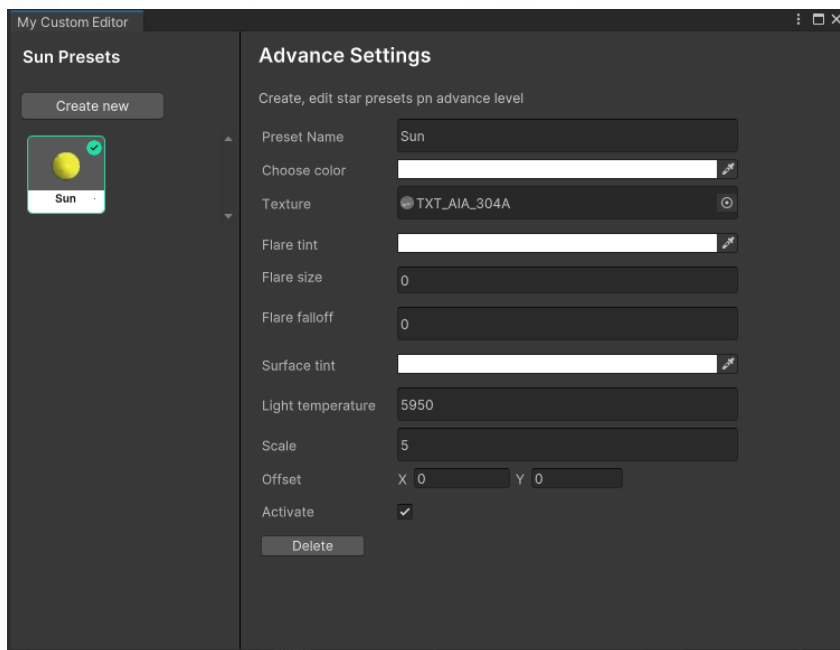
Texture: the texture that will appear on skybox.

Intensity: the light intensity in lumens.

Scale: the size of the star.

Offset : how far apart will be from the default star position.

For more star settings hit the "Advance settings" button.



Here you can edit settings like :

Flare tint : the color of star flare.

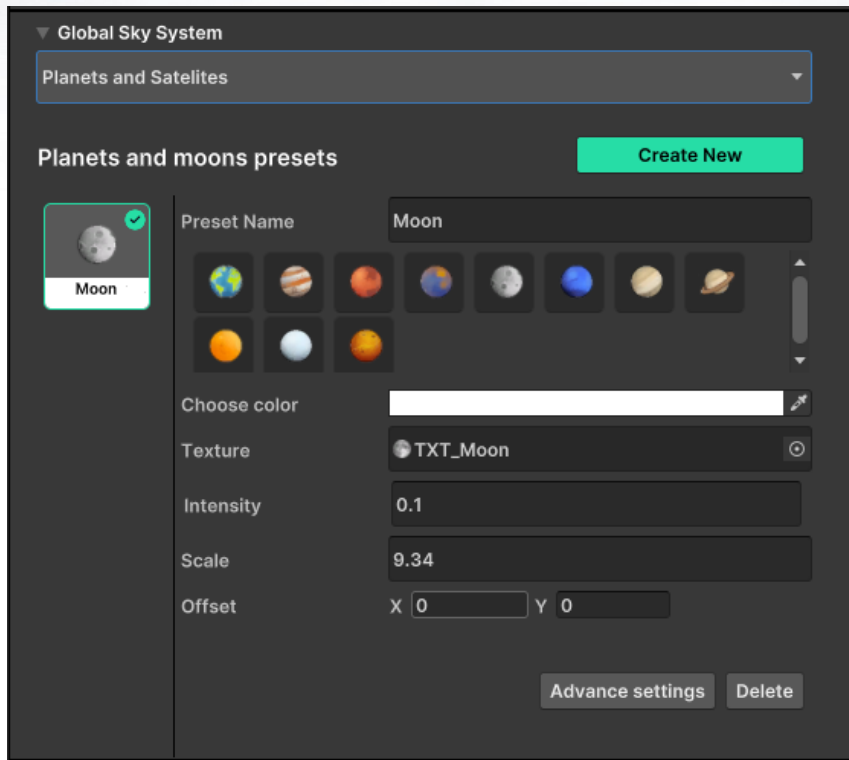
Flare size: the size of the star flare.

Surface tint: the color of star surface.

Light temperature: the light temperature measured in Kelvin.

Planets and satellites

Here you can create and edit presets for nearby planets and satellites:

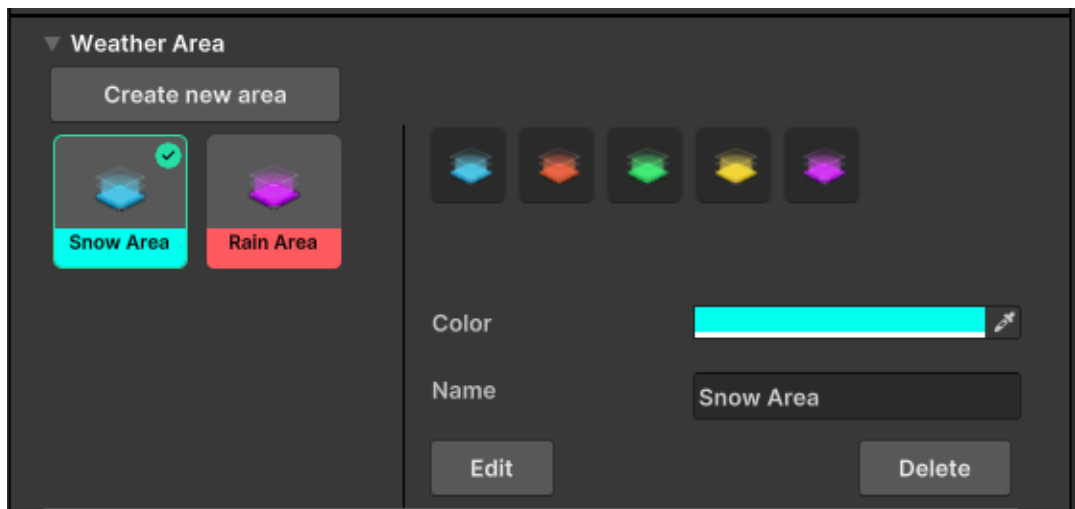


You have the ability to modify all the identical settings, just like the stars.

Weather Area

A "weather area" designates a specific zone characterized by a particular type of weather. As the player transitions between different areas, the weather will dynamically shift to correspond with the new location.

In the absence of designated "weather areas," the weather data will be universally applied across the entire scene.

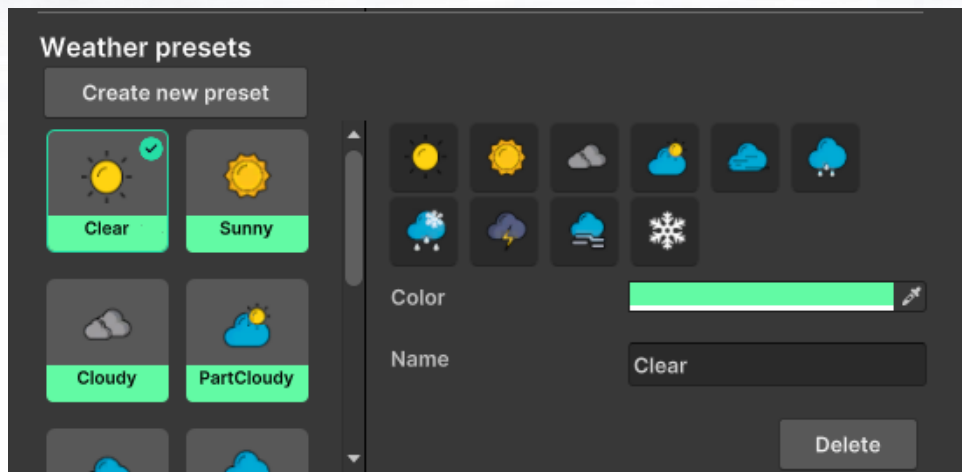


You have the option to select a personalized color and name for your area, enhancing its distinctiveness and facilitating quick identification.

Upon pressing the "Edit" button, the inspector will shift its focus to the weather area object. This enables you to relocate and adjust its scale, thereby defining both the location and dimensions of the weather area on your map.

Weather Presets

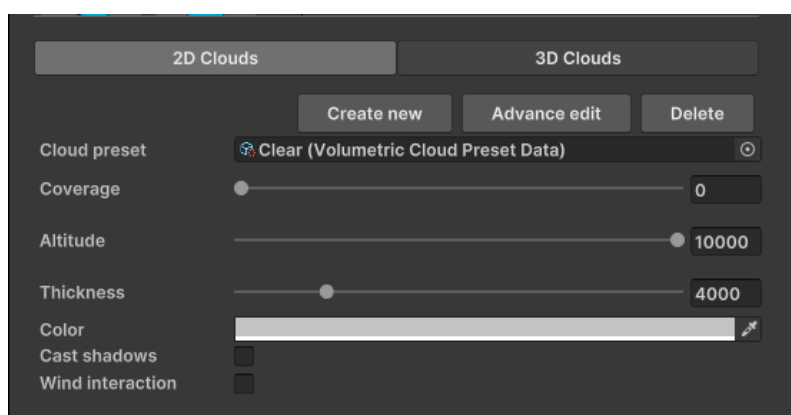
Weather presets encompass information pertaining to various weather elements, such as clouds and weather effects.



You have the option to select a personalized color, name and icon for your weather preset, enhancing its distinctiveness and facilitating quick identification.

Volumetric clouds

Within this interface, you have the capability to modify information concerning volumetric clouds.



Within the straightforward cloud editor, you have the capacity to alter elements such as:

Coverage: the density of clouds and the extent of sky coverage by clouds.

Altitude: the altitude in meters at which clouds are situated.

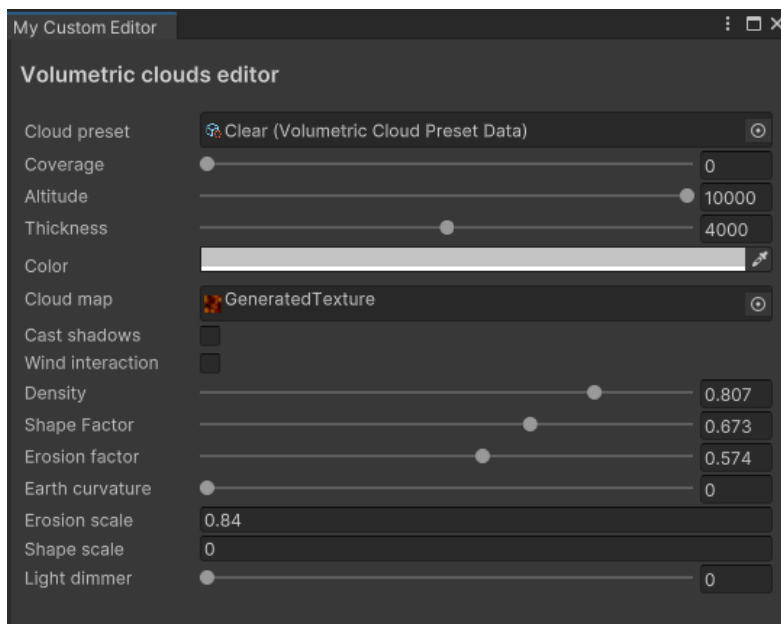
Thickness: the thickness of the clouds in meters.

Color: the color of the clouds.

Cast shadows: toggles if the clouds will cast shadows.

Wind interaction: enables or disables the influence of the global wind on the clouds.

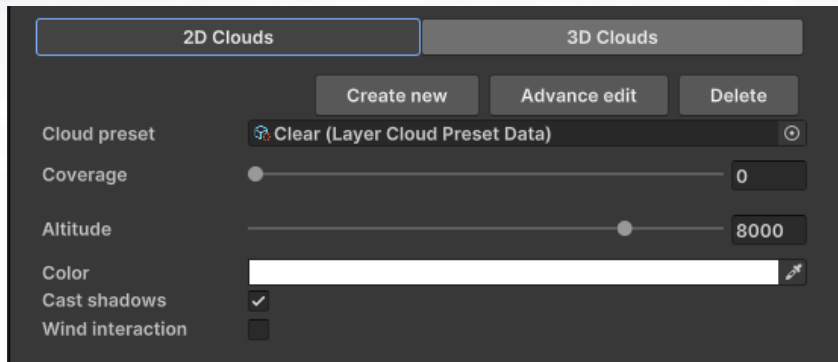
Volumetric clouds advanced editor



Within this section, you have the opportunity to modify additional parameters related to volumetric clouds. These parameters include the distribution pattern of clouds across the sky, the level of smoothness at the edges of the clouds, and the extent to which light is allowed to permeate through the clouds.

2D Clouds

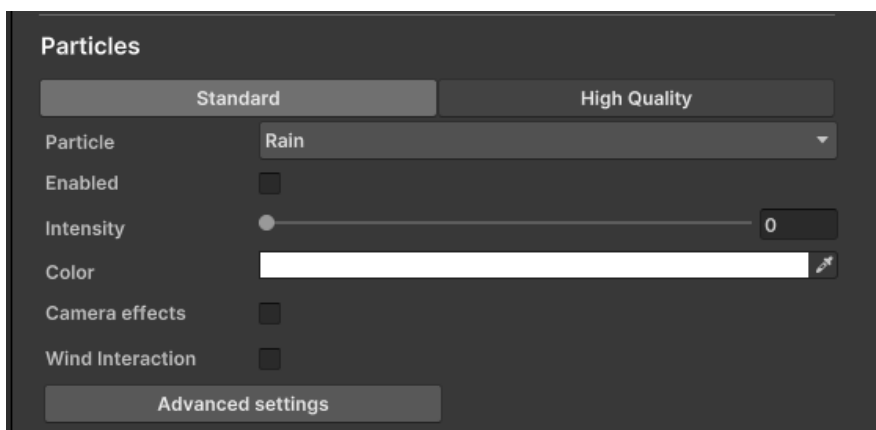
In this area, you can adjust parameters associated with the 2D clouds, including attributes like color, altitude, and coverage.



Particles

In this section, you have the ability to incorporate particles corresponding to different weather types, such as rain, snow, hail, and dust storms.

You can opt for either standard particles, which are created using the Unity Particle System and primarily rely on the CPU, or you can select high-quality particles generated with Unity VFX. The latter option is more GPU-oriented.



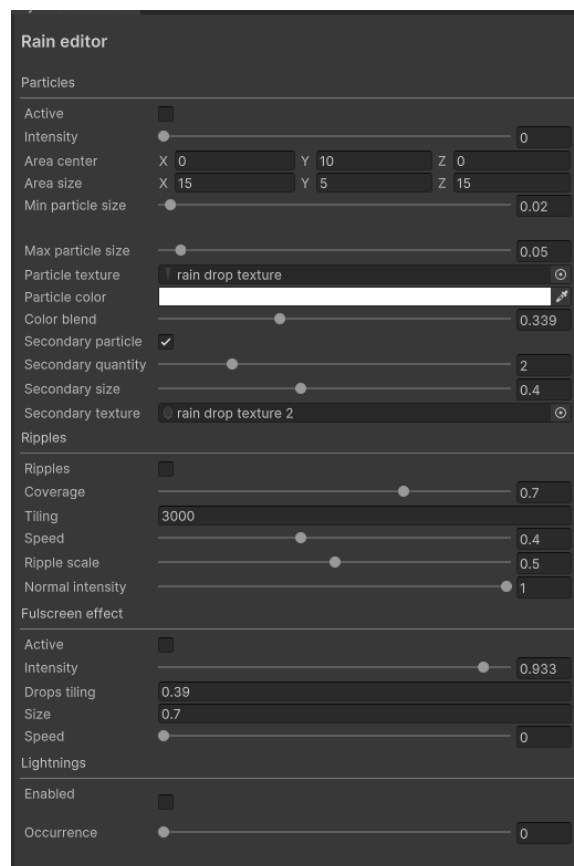
High quality particles

In the simplified editor, you have the choice between rain, snow, hail, fog, and dust storms. Within this interface, you can activate or deactivate particles, adjust their intensity, modify their color, and choose if the particles should be influenced by the wind.

For certain particles, there is an additional toggle labeled "Camera effects." Enabling this option will activate a full-screen effect associated with those particles.

Rain advanced editor

Within this specific section, you possess the capability to make more intricate adjustments to the rain settings.



Rain advanced editor

Parameters such as "Area size" and "Area center" play a crucial role in defining the dimensions of the zone where particles are visible and the offset between the camera and the particles.

Ripples

You have the option to enable or disable the appearance of rain ripples on the ground surface, and other related settings.

Full-screen effect

You can choose to enable or disable a full-screen effect featuring water droplets on the screen.

Lightnings

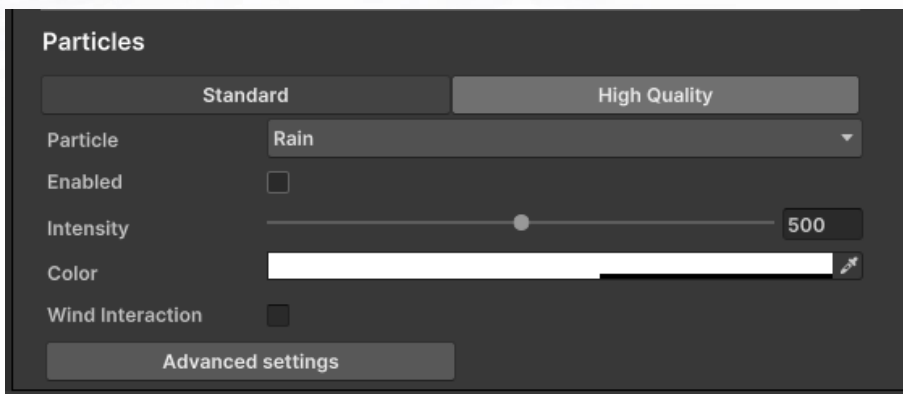
You can activate lightning and control their frequency, causing them to appear randomly around the camera's vicinity on the map.

Snow advanced editor

In addition to the settings available in the advanced rain editor, there is an additional "Snow layer" setting. Enabling this option will apply a layer of snow to the ground and on top of objects within the environment.

Standard particles

In the simplified editor, you can modify the selection of particles, enabling or disabling specific ones. You also have the ability to adjust parameters such as intensity, color, and the interaction of particles with the wind.



In the advanced editor, you can customize the area where particles are spawned and set the distance offset from the camera. Additionally, you have the option to modify the particle material and adjust the size of the particles.



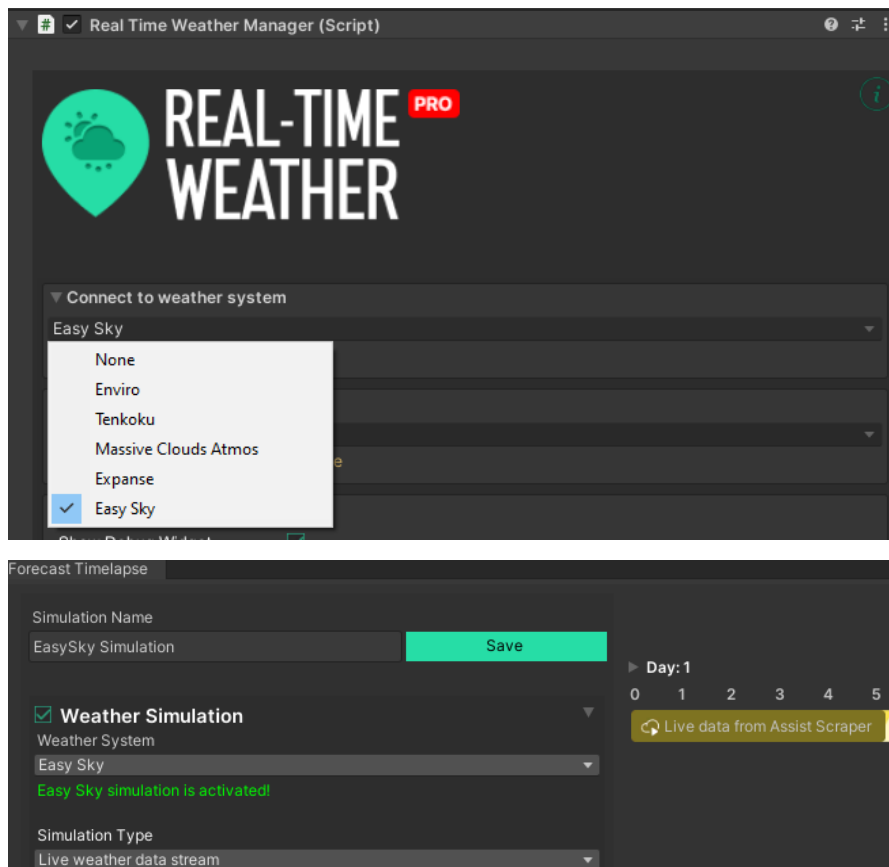
Unity Store Ready Integrations

Easy Sky is already integrated as a simulation tool for Real-Time Weather Manager PRO. [Unity Asset Store](#)

From simple weather data, to complex weather scenarios, Real-Time Weather PRO could be the perfect data provider (weather & maritime) to simulate weather simulations with Easy Sky.



While Real-Time Weather PRO is imported in your project, Easy Sky could be activated as the main weather system for the weather data obtained by Real-Time Weather PRO, using the provided main interface or from Timelapse Popup window, accessible while editing a weather simulation.



Scripting API

Easy Sky Weather Manager

The EasySkyWeatherManager class is implemented using the Singleton design pattern and it manages the main functionalities of the Easy Sky plugin.

It grants you the ability to govern every facet of weather, encompassing factors such as time, location, cloud presence, and weather effects.

Apply a weather preset:

```
private void SetWeatherPreset(WeatherPresetData presetData)
{
    CloudsController.SetLayerCloudPreset(presetData.LayerCloudPresetData)
;
    CloudsController.SetVolCloudPreset(presetData.VolumetricCloudPresetData
);
    WeatherEffectsController.RainController.ApplyData(presetData.RainData)
;
    WeatherEffectsController.SnowController.ApplyData(presetData.SnowData
);
    WeatherEffectsController.HailController.ApplyData(presetData.HailData
);
    WeatherEffectsController.DuststormController.ApplyData(presetData.Dust
stormData);
WeatherEffectsController.RainPsController.ApplyData(presetData.StandarRainData);
WeatherEffectsController.DuststormPsController.ApplyData(presetData.StandardDusts
tormData);
WeatherEffectsController.SnowPsController.ApplyData(presetData.StandarSnowData);
WeatherEffectsController.HailController.ApplyData(presetData.HailData);
}
```

Celestial Objects Controller

This class serves as a comprehensive solution for managing celestial objects within your virtual environment. From their seamless spawning to meticulous positioning across the sky, this functionality ensures that your celestial elements are precisely where they belong, enhancing the authenticity and immersion of your scene.

Solar Position Calculator

This class is a vital tool that accurately predicts the sun's position in the sky, leveraging input parameters such as date, time, and coordinates. By utilizing these factors, the calculator provides real-time calculations of the sun's azimuth and elevation, granting developers the ability to synchronize lighting, shadows, and overall ambiance in virtual environments with the dynamic movements of the sun.

Moon Position Calculator

This class is a crucial utility designed to precisely determine the moon's placement in the sky. By factoring in variables such as date, time, and coordinates, this calculator offers accurate calculations of the moon's azimuth and elevation. This functionality empowers developers to synchronize lunar positions with virtual scenes, enhancing visual authenticity and enabling dynamic moonlit atmospheres in their creations.

Clouds Controller

This class serves as the means to manage all aspects pertaining to clouds, encompassing attributes such as shape, altitude, thickness, and more.

To transition between different cloud types, utilize the "InterpolateBetweenCloudPresets" method, which facilitates a gradual shift from one cloud type to another.

Weather Effects Controller

This class functions as a comprehensive tool for overseeing all elements associated with weather effects, including rain, snow, hail, and more. It grants you the capability to regulate intensity, colors, and other parameters specific to each type of weather effect.

To transition between different weather types, utilize the "InterpolateEffects" method, which facilitates a gradual shift from one weather type to another.

Fullscreen Effects Controller

This class assumes control over the immersive fullscreen effects, specifically for rain and snow phenomena. It manages the activation, behavior, and customization of these atmospheric elements, enabling developers to effortlessly integrate and manipulate these weather effects throughout their scenes.

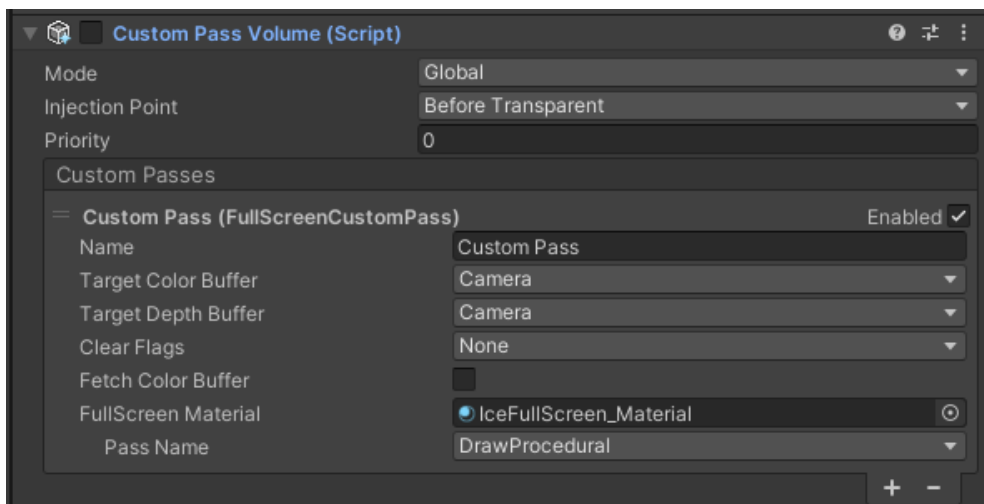
To transition between different full-screen effects, utilize the "InterpolateEffects" method, which facilitates a gradual shift from one effect to another.

Full-screen and Custom Pass effects

The implementation of full-screen effects relies on a specialized mechanism called Custom Pass, which augments the visual experience of elements like rain and snow. To create a Custom Pass, you incorporate a Custom Pass Volume component into your scene through the GameObject menu: `GameObject > Volume > Custom Pass`.

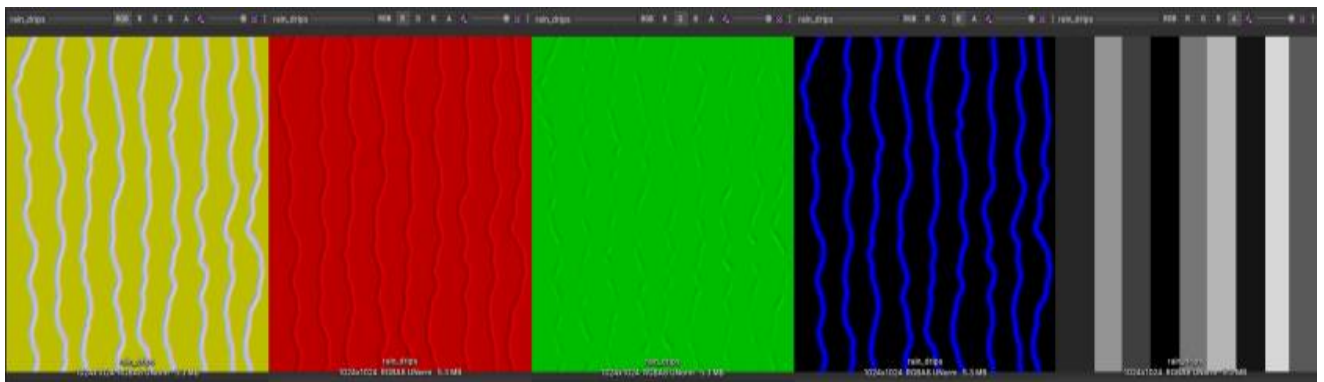
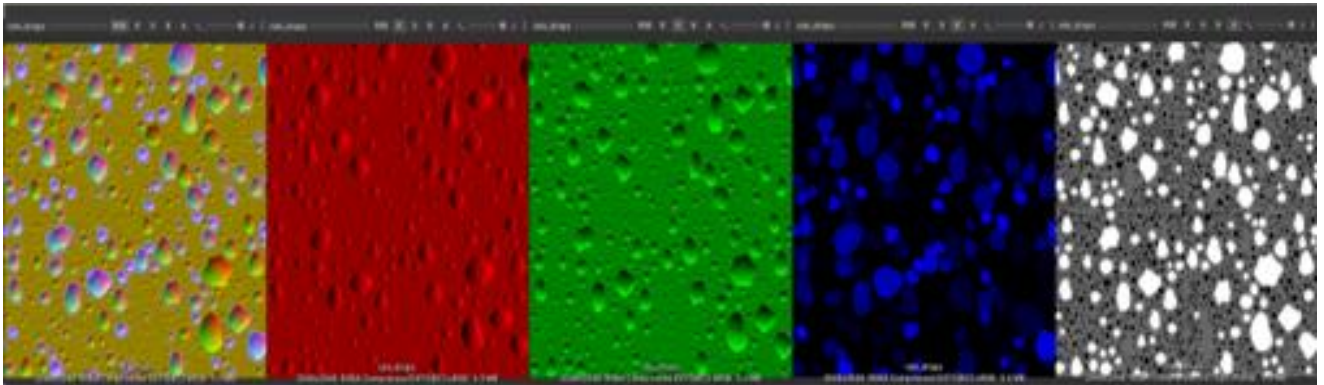
For the raindrop full-screen effect, a Custom Pass element is harnessed, configured with specific parameters. The "Target color depth" and "Target depth buffer" parameters are configured to correspond with the "camera" setting. Within this setup, a customized material derived from a dedicated shader designed for the effect is employed (refer to the provided image).

This custom material integrates settings for the full-screen effect. These settings reside within the material's surface inputs, offering controls over various aspects of the effect's appearance and behaviour. These encompass parameters such as the effect's intensity, tiling, size, and animation speed. By manipulating these inputs, developers can precisely tailor the full-screen raindrop effect to achieve their desired visual impact.



The raindrop and drip-on-lens Full-screen shader uses three textures. Two textures are made with channel packing, each channel (RGBA) containing different information. The raindrop image contains normal data in the red and green channels, a mask for animation (temporal offset) in the blue channel, and a mask for animated elements in the alpha channel. In the alpha channel, the white area represents animated elements.

The rain-drip image contains normal data in the red and green channels, a mask for the drip path in the blue channel, and a mask for animation (temporal offset) in the alpha channel.



The full-screen ice effect operates on a distinct shader tailored specifically for icy visuals. The Custom Pass approach is maintained, with identical settings for the "Target color depth" and "Target depth buffer" parameters.

The unique facet of the full-screen ice effect emerges from its specialized ice shader. Within the shader, the material's surface inputs host a suite of adjustable parameters for managing the effect's characteristics. These parameters encompass:

- 1. Effect Intensity:** Governs the strength of the ice effect. Adjusting this parameter allows developers to modulate the prominence of the icy overlay on the screen.
- 2. Screen Coverage:** Dictates the extent to which the icy texture spans across the screen. By modifying this input, developers can control the distribution of the ice effect.
- 3. Texture and Noise Fringe:** This component introduces a noise texture that disrupts the outline of the effect, mimicking the natural fragmentation of ice. By manipulating this parameter, developers can influence the realism and complexity of the ice's appearance.
- 4. Intensity Parameter:** This parameter serves a dual purpose: it regulates both the transparency of the texture and the degree of distortion imparted by the effect. Adjusting this input empowers developers to manage both visual clarity and the subtle, dynamic distortion that characterizes ice formations.

Through the careful manipulation of these surface inputs, developers can craft an authentic and immersive full-screen ice effect that adheres to their creative vision, contributing an extra layer of depth and realism to their scenes.

The rain ripples effect was meticulously crafted using a custom pass technique, utilizing a ripple shader as its foundation. This approach yields a dynamic and realistic simulation of raindrop ripples across surfaces. The effect's behavior and appearance can be tailored using a set of adjustable parameters, ensuring versatility in creating convincing rain-affected visuals.

The following parameters are exposed and configurable within the ripple shader:

1. **Ripple Coverage:** Dictates the density of the ripples, allowing developers to control the frequency at which raindrop ripples appear on surfaces. Adjusting this parameter lets you modulate the overall intensity of the raindrop impact.
2. **Top Coverage:** Determines the extent to which the ripples are visible on horizontal surfaces, considering the angle of the surface. This parameter grants developers the flexibility to influence how far the ripples extend from horizontal surfaces.
3. **Tiling:** Governs the size of the ripple groups, contributing to the spacing and arrangement of the raindrop ripples on surfaces.
4. **Ripple Scale:** Modifies the scale of individual ripples. This parameter enables you to adjust the size of each individual raindrop ripple, further enhancing the realism of the effect.
5. **Normal Intensity:** Controls the strength of the normal map, affecting the prominence of surface deformations caused by the raindrop ripples. Adjusting this parameter impacts the visual depth and dimensionality of the ripples.
6. **Speed :** Controls the speed of the animation.

By manipulating these exposed parameters, developers can finely tune the rain ripples effect to harmonize with the specific ambiance and style of their virtual scenes. The result is a visually engaging raindrop ripple simulation that enhances the overall visual fidelity of the environment.

The snow effect employs a dedicated Custom Pass element in conjunction with a tailored snow material. This approach ensures a visually compelling snowfall experience within the scene. The customizable material settings provide developers with a range of options to fine-tune the appearance and behavior of the snow effect.

The snow material encompasses the following adjustable settings:

1. **Amount:** Controls the density and intensity of the snowfall. By manipulating this parameter, developers can simulate light flurries or heavy snowstorms.
2. **Direction:** Dictates the angle at which the snowflakes descend. This parameter contributes to the sense of movement and wind within the snowfall.
3. **Patch Size:** Regulates the size of individual snow patches. Adjusting this parameter influences the scale of the snow accumulation effect.
4. **Patch Fringing:** Governs the irregular edges of snow patches. By introducing fringing, developers can simulate the natural unevenness that occurs as snow accumulates.
5. **Texture and Tiling:** The texture applied to the snow effect and its tiling factor. These parameters offer control over the visual characteristics of the snowflakes themselves.
6. **Color Tint:** Enables developers to modify the hue of the snowfall, allowing for customization to match different scene aesthetics.
7. **Specular Shine:** Controls the specular reflection of light on the snowflakes, affecting their sheen and highlighting.

The formation of snow patches is an integral aspect of the effect. When the coverage area isn't 100%, these patches are generated. The shape and fringing of these patches are influenced by noise functions. A larger noise function contributes to patch shape variation, while a smaller one creates fringing that disrupts the smooth outline of the patches, simulating the natural unevenness of accumulated snow.

By adjusting these material settings, developers can meticulously craft the appearance and behavior of the snowfall, creating a visually captivating and immersive wintry atmosphere within their scenes.

Natural Phenomena Particle System

To comply to different hardware capabilities, we've employed a versatile approach for particle systems – leveraging the VFX graph for high-end devices and the older particle system (previously known as Shuriken) for low-end and mobile devices. This dual-pronged strategy ensures optimal performance across a wide spectrum of devices.

For the advanced VFX graph, we've organized the exposed parameters into three distinct groups, streamlining the workflow for developers:

- 1. Spawn Parameters:** This group encompasses settings related to particle generation, controlling the rate and behavior of particle emission. These parameters allow developers to finely adjust the density and timing of particle spawn.
- 2. Particle Size & Color:** Here, developers can manipulate the size and color characteristics of the particles. This includes controlling the scale and appearance of the emitted particles, ensuring a cohesive visual aesthetic.
- 3. Physics Parameters:** The physics group encompasses settings that govern the behavior of particles over time. This includes attributes like particle velocity, rotation, and response to external forces, enabling developers to achieve the desired physical interactivity within the particle system.

As an illustrative example, when using flip-book images for elements like snow, hails, and dust storms, each image operates as a flip-book animation. Within these animations, grids define different aspects like size and positioning, thus influencing the flip-book's dimensions.

This thoughtful approach allows developers to optimize particle systems for both performance and visual quality, accommodating a diverse range of devices while maintaining a cohesive and immersive experience across the board.



FAQs

Q: I don't know how to use the Easy Sky plugin. Do you have a tutorial?

A: Yes, you can check out our tutorials from Unity Asset Store product page. You can also read the **Overview & Installation** chapter from the documentation, which has all the information you need for any setup.

Q: Which pipeline renderers are supported?

A: Easy Sky Weather Manager works on HDRP only.