

Sky Master ULTIMATE Setup Guide

Overview

The system is a complete Sky, Volumetric Clouds, Atmosphere, Lighting, Ocean - Water and Real Time Global Illumination system, combining all together for minimal extra work to make all modules work together.

The core system is made for the Standard Pipeline and **URP and HDRP Betas are available to all users** of the system on request. Note that URP and HDRP versions are not only conversions, but a complete remake of the system, adding several new and exciting features, like volumetric lights in the fog in URP and a cutting edge new water system in HDRP.

Contact me directly for any help with the setup described in the following and for the Betas of URP and HDRP versions. ([Discord](#), [Email](#))

The volumetric clouds have three types, for various uses and performance gains.

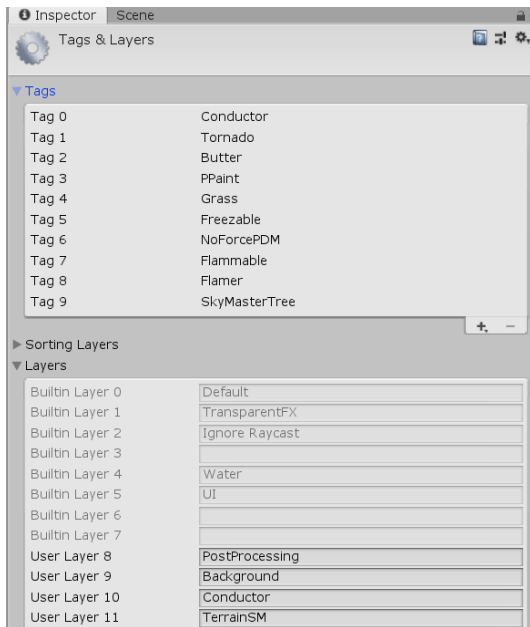
- **Full Volumetric Clouds:** There is the latest system using image effect on the whole scene and allow fly through the clouds and planet like effects ("CloudScript" component on Camera). This system can be combined with the custom Temporal AA included in the package to reduce the noise artifacts from cloud sampling ("Cloud Temporal Anti Aliasing", "Velocity buffer" and "Frustrum Jitter" scripts on camera)
- **Background volumetric clouds:** The second system is also created as image effect ("FullVolumeCloudsSkyMaster" script on camera), but is only applied to the background of the scene for extra performance gains, by splitting the calculations in multiple frames and downscaling. This system does not support fly trough and requires an extra camera rendering objects that will be behind the clouds (e.g. moon, galaxy and stars are already setup for the system in the latest prefabs)
- **Mesh based volume clouds:** The third system is shader based volumetric, using multiple mesh layers to render the volume ("CloudHandlerSM" script on "SHADER VOLUME CLOUDS" gameobject manages the mesh layers and the cloud properties). This system has the advantage that acts like any other scene gameobject and is not based on image effects, so can easily be reflected on any water system and can have items behind the clouds, also can support look from below or above the clouds, though does not support fly through.

The system also has a simple shader based cloud system for use on lower end mobile and as extra cloud layer to the volumetric clouds above ("DOME CLOUDS L1-L2" gameobjects under the "SKY MASTER MANAGER" gameobject).

Setup steps

To setup the system, there is two different methods, one is through dragging a prefab in the scene and the other step by step adding the sky and all needed components. Each method has its own advantages.

Before the setup a few layers and tag should be added to handle the various system functionalities.



The main layers and tags to setup are the **“Background”** layer which is used for objects rendered behind the clouds, the **“Conductor”** tag is used to make objects attract lightning from the Lightning system and the **“TerrainSM”** layer which is used to limit to certain objects the top down depth rendering that is used in the shore line system in the water module.

The various other tags are used in the advanced particle and collisions systems that are detailed in the script references manual and are not directly used by the core weather system.

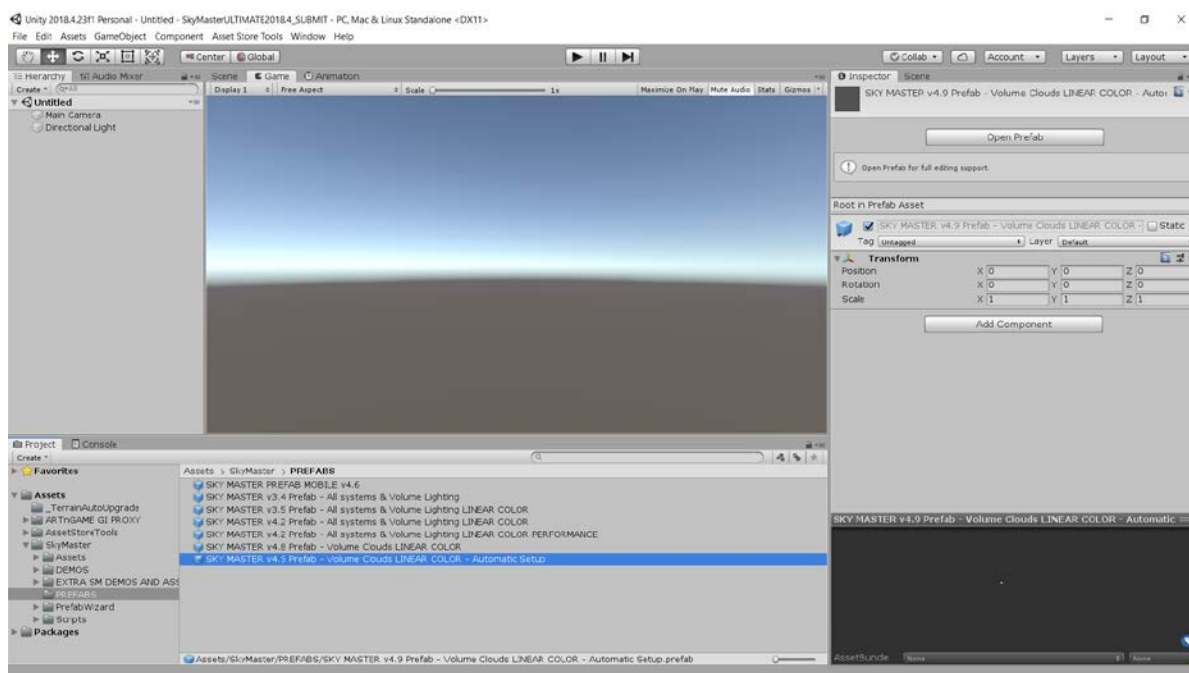
NOTE: The player is by default the camera tagged as **“MainCamera”** and one such camera must exist in the scene for the system to function properly.

Prefab based setup

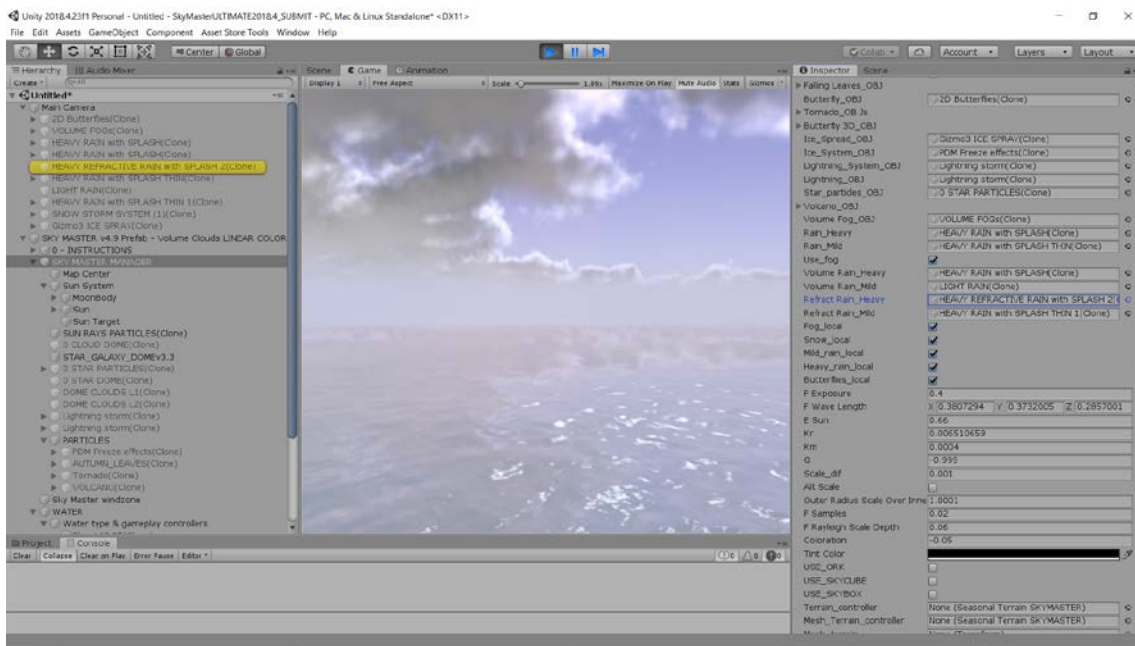
With the prefab, the system auto configures all the components needed to the existing scene camera and every module is available and working together with the rest and can be tweaked or some components can be disabled for later use. This is the recommended method and easiest to get started with. (Refer to [Tutorial Video Prefab Setup and Tweak](#)).

The prefab contains the second cloud type as default, the others can be added manually for the latest first type by adding the “CloudScript” script on camera or by the Sky Manager Configurator in the mesh volume clouds case.

First open the scene that requires the sky, then locate the prefab named “SKY MASTER v4.9 Prefab - Volume Clouds LINEAR COLOR - Automatic Setup” in the “SkyMaster” ---> “PREFABS” folder.



Then **drag the prefab in the scene to enable the sky system** and erase the directional light, as the system adds a Sun light on its own.



The system applies lightning by instantiating a lightning system at run time, when the relevant option is enabled (default) in the FullVolumeCloudsSkyMaster script (“Enable Lightning” checkbox). Also if the “Use Weather” checkbox is enabled, the lightning will start and stop based on the chosen weather type in the Configurator.

The **“Volume Clouds Lightning Bounding Box”** defines the region where the instantiated lightning modules will randomly appear in and can be tweaked or moved to get them to the required place.

The **“Volume Clouds BackLayer Camera”** is used to render the object behind the clouds when the “Blend Background” option is activated in the clouds script, using the “Background” layer. All objects needed behind clouds, e.g. “MoonBody” for the moon, “STAR_GALAXY_DOMEv3.3” for the galaxy shading and “0 STAR PARTICLES” for the particle based stars are in that layer by default in the prefab.

The **“Volume Clouds Shadow Dome”** casts shadow from the clouds and can be scaled in the case the camera view distance is not enough to reach the dome with the shadows shader.

The various Configurator options and system usage will be presented below in the step by step configuration. The prefab has most of the steps described in the step by step procedure already implemented, these steps **will be colored in blue for clarity**.

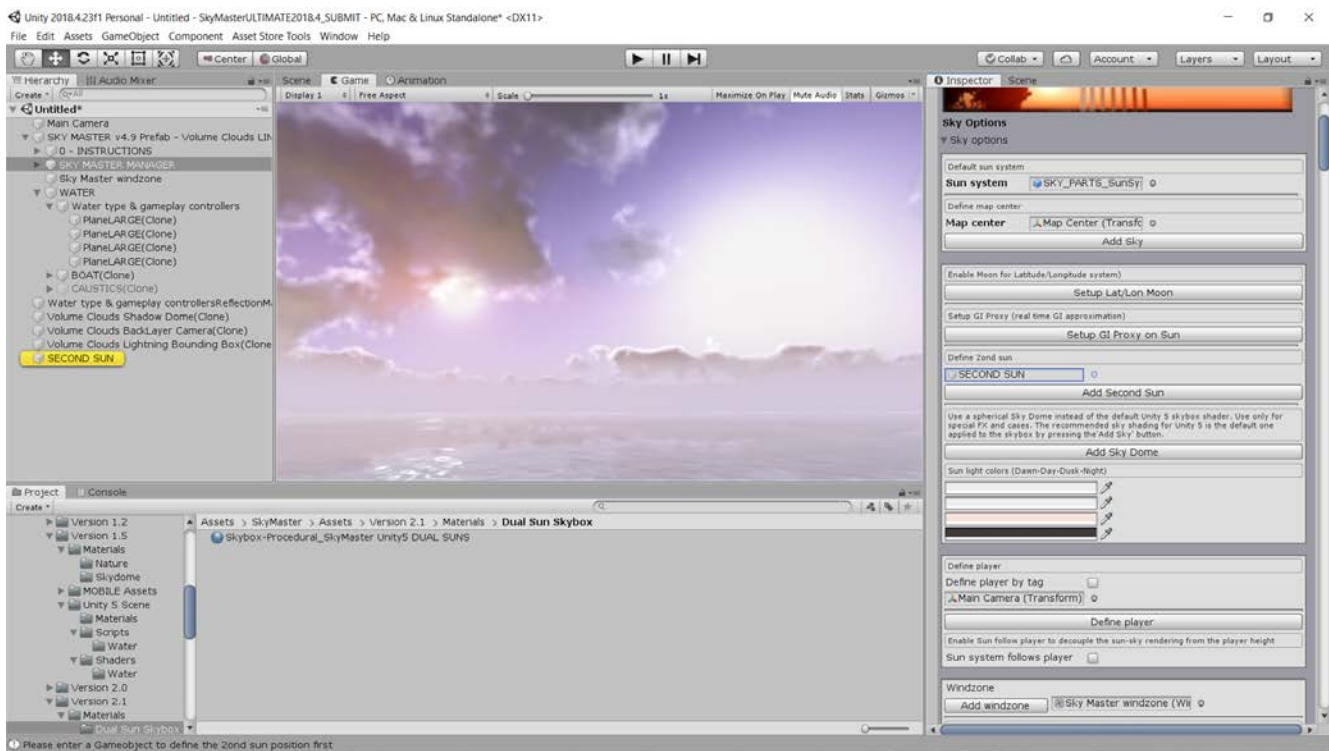
Step by step based setup (Advanced)

1. Sky

For the step by step setup, **first add the “SkyManager” component to an empty gameobject in the scene, preferably named as “SKY MANAGER”**. This will add the **Sky Master Configurator** inspector.

Press the “Add Sky” button to add the main sky to the scene, this is a material that is assigned in the Rendering - Lighting – Skybox Material section in the Unity Lighting settings.

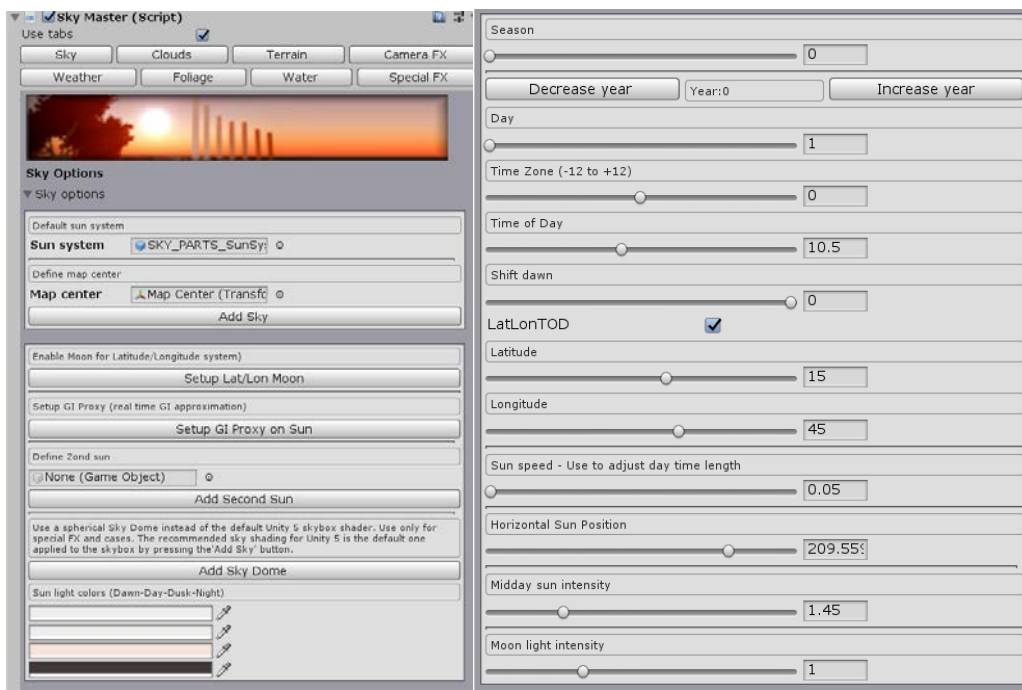
The **“Add Second Sun”** button changes to the skybox shader with a 2ond sun support with the full atmospheric scattering model, to define the 2ond sun position must have an object on scene and reference it in the “Define 2ond sun” slot in the manager.



By default, the system sun controller is the simple version where the horizontal and vertical angles are directly controlled by the **“Time of Day”** slider (vertical) and **“Horizontal Sun Position”** slider (horizontal). To enable the Latitude-Longitude based sun positioning, press the **“Setup Lat/Lon Moon”** button, this will setup the moon in the system and enable the **“LatLonTOD”** option where the control of the sun position is now defined by the **“Time of Day”** slider, **“Time Zone”** and **“Latitude”**, **“Longitude”** sliders.

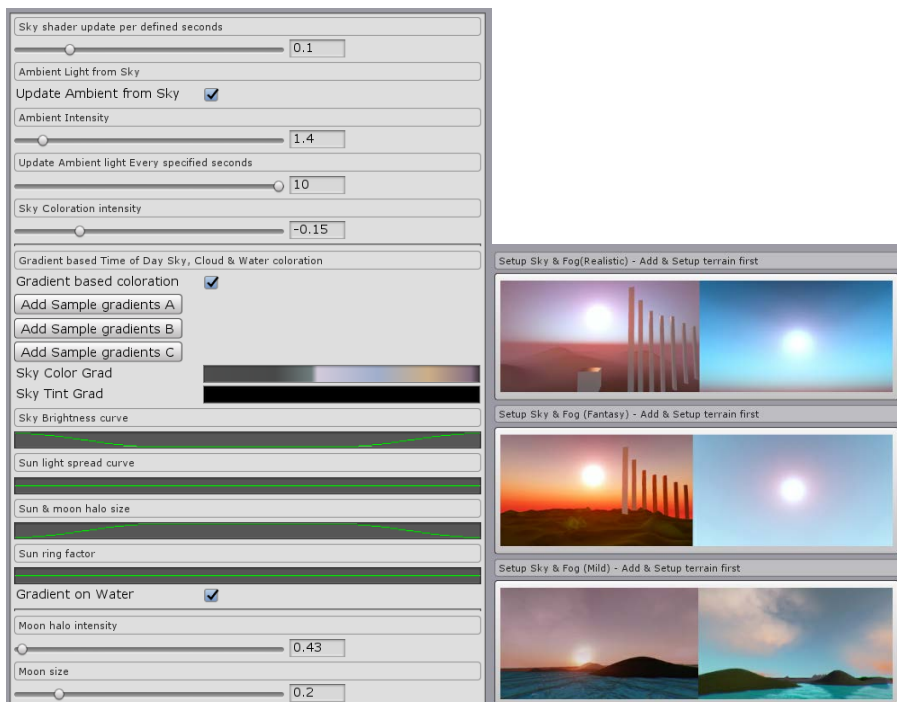
The speed of day change is defined using the **“Sun Speed”** slider and the lights intensity by the **“Midday Sun Intensity”** slider for sun light and **“Moon light Intensity”** slider for moon light.

API: To change the time of day and day cycle speed during gameplay, set the **“Current_Time”** and **“SPEED”** variables of the SkyMasterManager script respectively. A sample of the API is provided in the latest **“Demo scene v5.0 MAIN DEMO and API EXAMPLE SCENE”** demo scene, in the **“API EXAMPLES CONTROLLER”** gameobject and **“SKYMASTER_API_5_0”** script.



NOTE: Before changing the sun position mode, first right click on the prefab in the Hierarchy and select the **“Unpack prefab completely”** option, so can be directly edited by the sky manager.

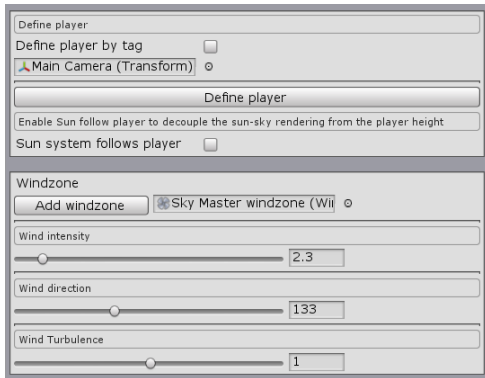
The sky rendering control defaults to the presets system, where the sky properties are hardcoded in the SkyMasterManager code and the various presets can be accessed through the icons shown below or the **“Preset”** variable in the SkyMasterManager script.



The latest version has a new sky rendering control method, using curves and Gradients, for more intuitive and not hard coded way to control the sky. This option is enabled using the **“Gradient based Coloration”** checkbox and the curves and gradients that control the sky properties are available to fine tune the sky look.

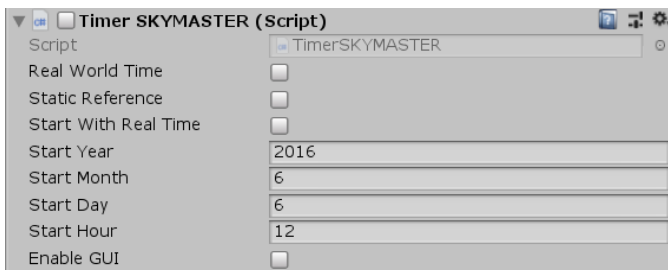
The moon size and halo can be controlled using the **“Moon size”** and **“Moon halo intensity”** sliders.

2. Player and Wind zone options



The player is by default the camera tagged as **“MainCamera”**, and all particles and systems will follow this by default. The player can be changed using the **“Define player”** button, after dragging a Player object in the slot above the button or by tag using the **“Player”** tag on an object and enable the **“Define player by tag”** checkbox. The tag is **“Player”** by default, but can be changed in the **“Player_tag”** variable in the SkyMasterManager script.

The system uses a wind zone to apply directionality to cloud motion, [so a windzone should be created using the “Add windzone” button](#) and can be controlled with the relevant sliders in the windzone section.



The time used in the default core system is the Unity time, to use a more precise time model a script **“TimerSKYMASTER”** is provided, the script must be in the same gameobject as the SkyMasterManager script and enabled. The script allows to use an exact time based on real world clock and precision time with a starting reference date.

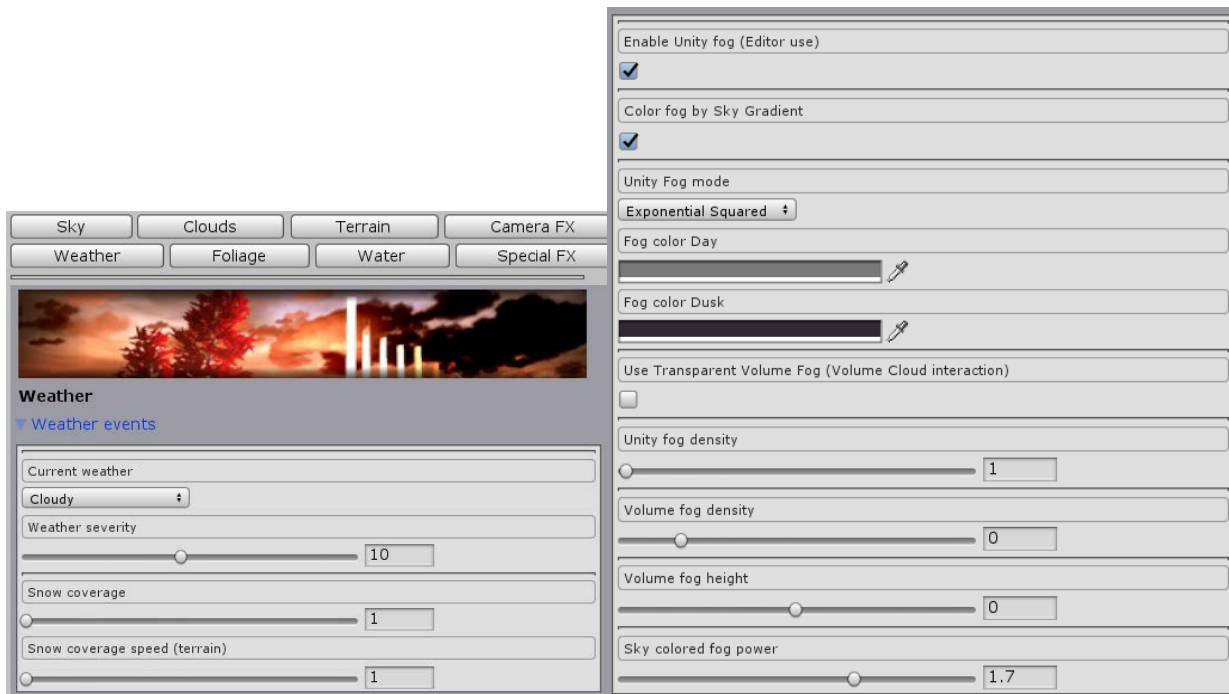
3. Weather configuration

In the weather section in the configurator, the various weather types can be selected and enabled using the **“Current weather”** drop down menu. The **“Weather Severity”** slider can adjust the particle count from very low, to heavy for the rain and snow effects. The **“Snow coverage”** sliders apply a global shader variable change that increases the snow amount in the snow shaders. The system offers two snow shaders, one for meshes and one for Unity terrain. The snow shaders can be setup in the Terrain section or manually by applying the snow materials to the mesh or Unity terrains.

The **“Enable Unity fog”** checkbox can be used to enable the unity fog. The mode of the fog can be set in the **“Unity Fog Mode”** section drop down menu and the colors in the **“Fog Color Day-Dusk”** sections. The density – height – power sliders can be used to fine tune the fog effect.

NOTE: The Unity fog is applied only in Forward rendering mode, in Deferred mode the gradient volumetric fog (“GlobalFogSkyMaster” script on Camera) image effect is used (can be setup in the “Camera FX” section or comes pre setup if the prefab is used).

The **“Color fog by Sky Gradient”** can be used to override the volumetric fog coloration based on the sky colors in various times of day.



Weather events can be programmed in the menu shown below, use the **“Add event”** button to add a weather change event after specifying the weather type and start and end dates. An event can be removed by pressing the minus **“-”** button on the right side of the event.

Add event

Weather events have two modes, one is specific day definition and the other the cycle mode where the month and day in the month is specified and the event will repeat every year the specific date. In non loop mode the month is not used-required and day should be calculated from game start (e.g for January 1st in 20nd year is 365 plus 1 days and the event will trigger only once at that time).

Chance	Time span	Day span	Month spar	Weather	Loop	E
50	1-24	1-30	1-2	LightningSt	False	-
50	1-24	1-30	11-12	LightningSt	False	-

Weather type: Lightning Storm

Weather Follow: Cloudy

Loop weather: ☐

Weather chance (%): 50

Start hour (1-24): 1

Start day (1-30 for loop): 1

Start month (1-12 for loop): 11

End hour (1-24): 24

End day (1-30 for loop): 30

End month (1-12 for loop): 12

Volume cloud height: 100

Volume cloud span: 1000

A weather randomizer sample script is also provided as showcase of the API and to allow to get randomly changing weather patterns at a specified interval.

Weather Randomizer SKYMASTER (Script)

Script: WeatherRandomizerSKYMASTER

Enable GUI: ☐

Affect Water: ☐

Affect Fog: ☐

Affect Fog Params: ☐

Change Weather Interval: 300

Current Weather:

Cloud Density Change Spec: 1

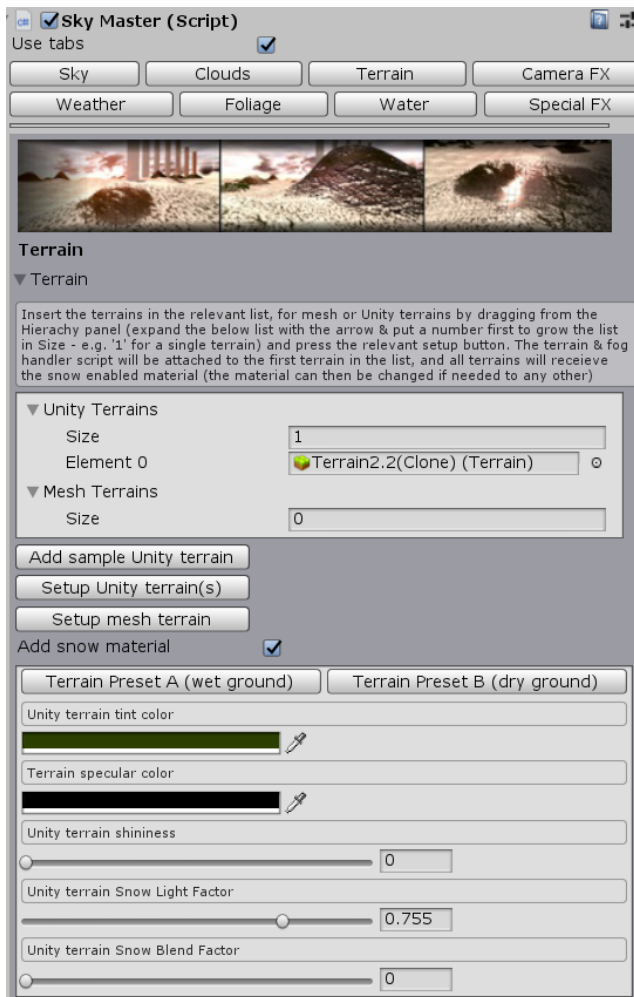
API: To change the weather during gameplay, set the “**CurrentWeatherName**” variable of the SkyMasterManager script. A sample of the API is provided in the latest “**Demo scene v5.0 MAIN DEMO and API EXAMPLE SCENE**” demo scene, in the “**API EXAMPLES CONTROLLER**” gameobject and “**SKYMASTER_API_5_0**” script.

4. Terrain configuration

The system offers the option to setup a Unity or Mesh terrain with a special material that allows snow to gather gradually when the snow weather is activated.

The snow accumulation and density parameters are in the Weather section mentioned in the previous section.

In order to setup the snow shader in the mesh or Unity terrain, first insert the terrain in the proper list and then press the “Setup Unity-Mesh terrain” button. The “Add snow material” checkbox must be activated for the material to be applied to the terrains.



The material for meshes can also be applied to any mesh that requires snow to gather on it and the snow amount is controlled by a global shader variable for all related materials automatically.

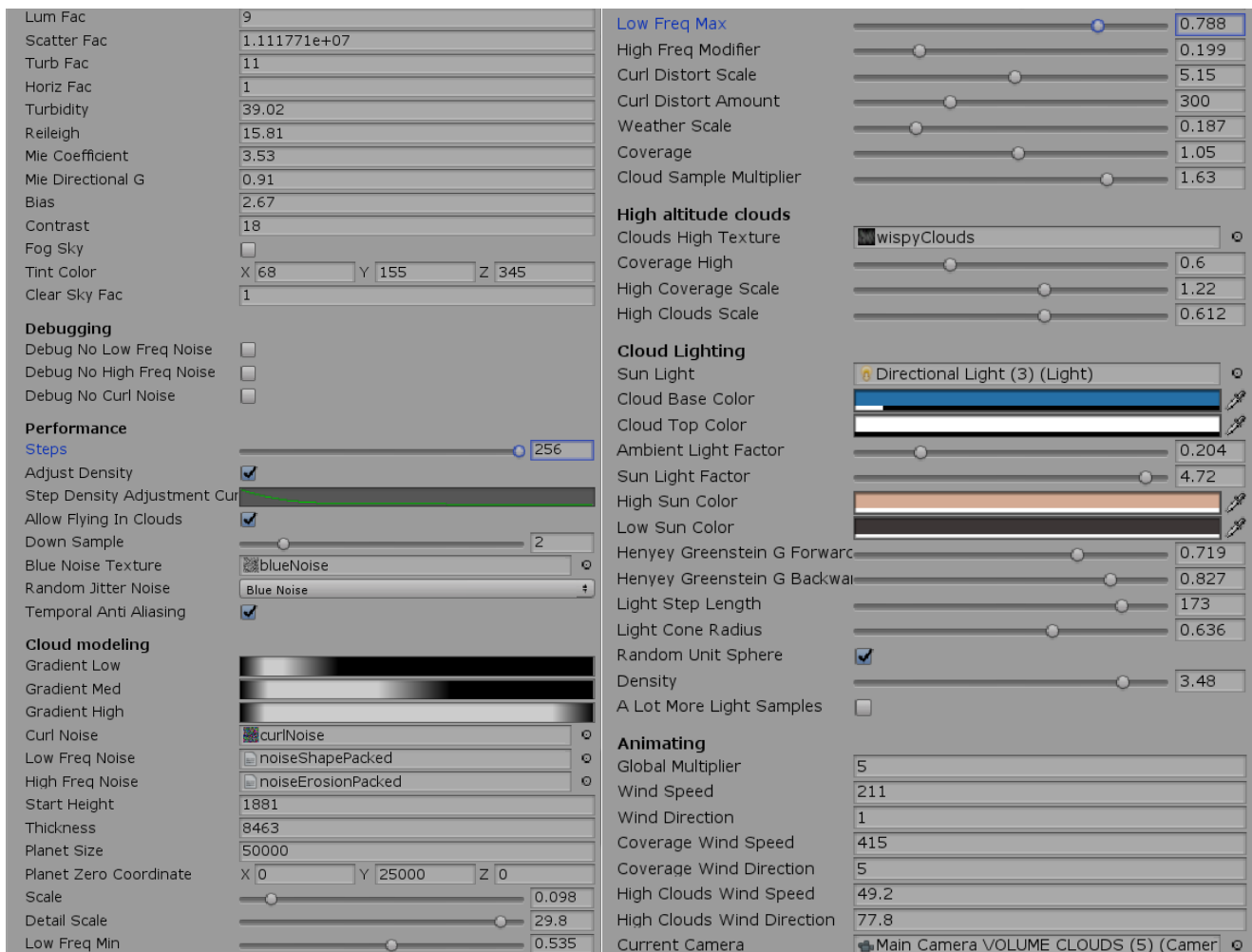
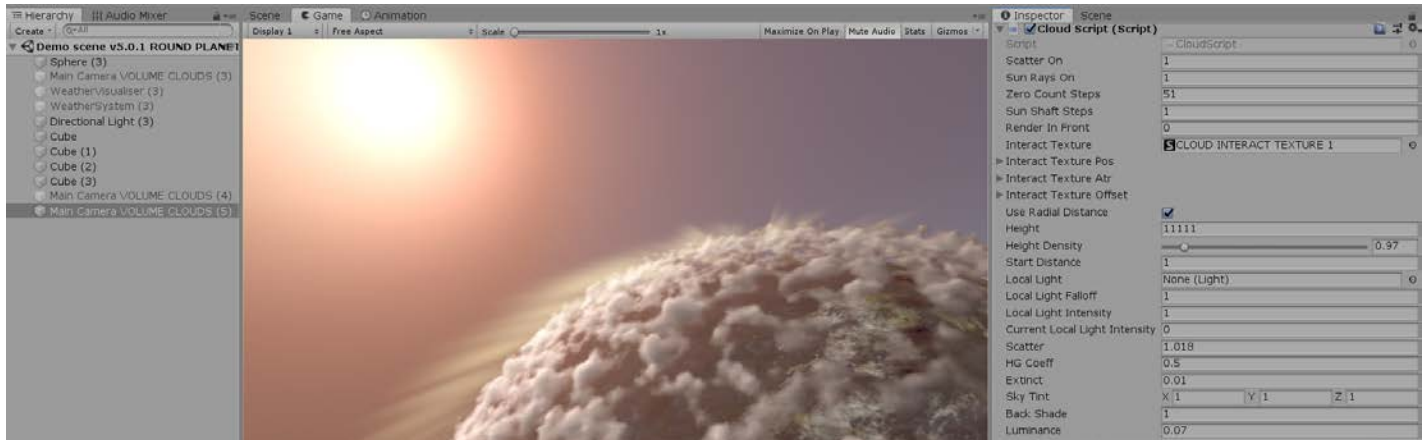
This system is also directly compatible with the InfiniGRASS asset grass and trees, so can grow snow when used with Sky Master UTLIMATE.

5. Clouds configuration

The various cloud types can be setup from the configurator or directly on the camera. The latest Full Volumetric clouds can only be setup directly on the camera.

5a. Full Image Effect Volumetric Clouds

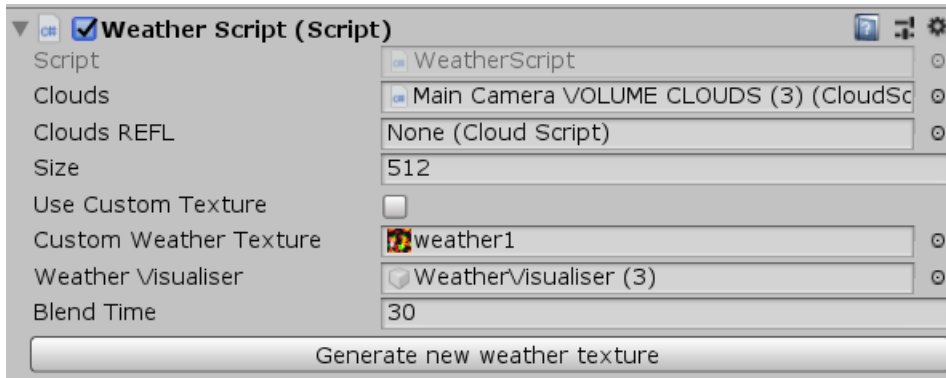
These clouds support fly through and are implemented as an image effect on the camera. In order to setup the system, add the component named **“CloudScript”** on the main camera of the game. The system will auto configure the Temporal AA it uses to cover the sampling artifacts.



The next setup step is to add the sun light to the “Sun Light” slot, so the system can regulate the cloud lighting based on the sun. If Sky Master manager is setup in the scene, this light should be the Sun light under the “SKY MASTER MANAGER”-> “Sun System” gameobjects.

Use the “**Planet size**” variable to adjust how far and curved in horizon clouds will be and the “**coverage**” to make the clouds thicker or sparser.

The system creates clouds based on a generated weather map, using the following module (“**WeatherScript**” script).



To setup the passing of the texture to the cloud system, the system must be referenced in the “**Clouds**” slot of the script (Drag the Camera that has the “**CloudsScript**” script to the slot from in the Hierarchy to the script Inspector “**Clouds**” slot). If “**Use custom texture**” is selected, the weather map can be manually defined in the “**Custom weather texture**” slot.

5b. Background Image Effect Volumetric Clouds

These clouds are rendered in the background behind scene objects and are implemented as an image effect on the camera. To setup the system, press the “Create Volumetric Clouds v4.8” button in the Configurator, in “Clouds” section.



The volumetric clouds system **allows to set up the water reflection camera to render reflection of the clouds** if the water module is used and one that will render objects that need to be placed behind the clouds.

To add the system to the water reflection camera for the water module, press the **“Create Reflections on Sky Master Water”** button and the **“FullVolumeCloudsSkyMaster”** script will be added to the reflections camera and be setup with the **“IsForReflections”** option.

In order to render objects behind the clouds (e.g. Moon, Galaxy, Stars), use the **“Setup Back Layer”** button. The system assumes that a layer named **“Background”** has been created beforehand and that objects needed to be behind the clouds are assigned to this layer.

A lightning system is also available and can be setup using the **“Setup Lightning”** button, this will create the bounding box around which lightning will appear and link the lightning system to the clouds script. The lightning is activated on Lightning Storm weather if **“UseWeather”** option is enabled in the clouds script on camera, or can be manually enabled by enabling the **“EnableLightning”** checkbox on the clouds script.

Shadows from the clouds may be setup using the “Setup Shadows” button, this will add a mesh dome in the scene that will cast shadows using a shader in sync with the volumetric clouds rendering.



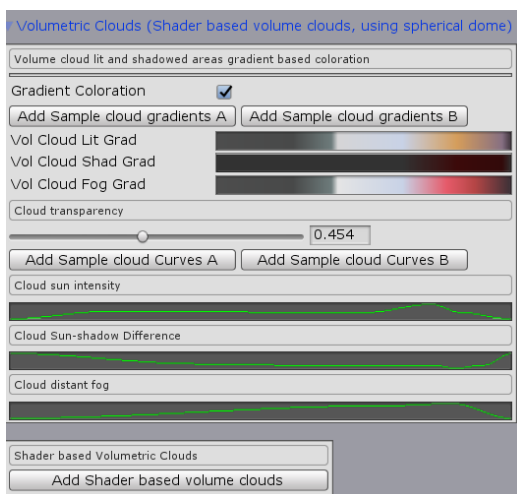
The buttons **“Move Volume Fog to Top –Bottom”** will move the script in the rendering order in the camera, to be above or below other image effects and the volumetric fog. This is useful when need to regulate how the cloud volume blends with the volumetric fog.

NOTE: The **“Cloud density – Sunny - Cloudy – Storm”** weather types will be applied in the first weather change that happens after the game starts.

5c. Mesh based Volumetric Clouds

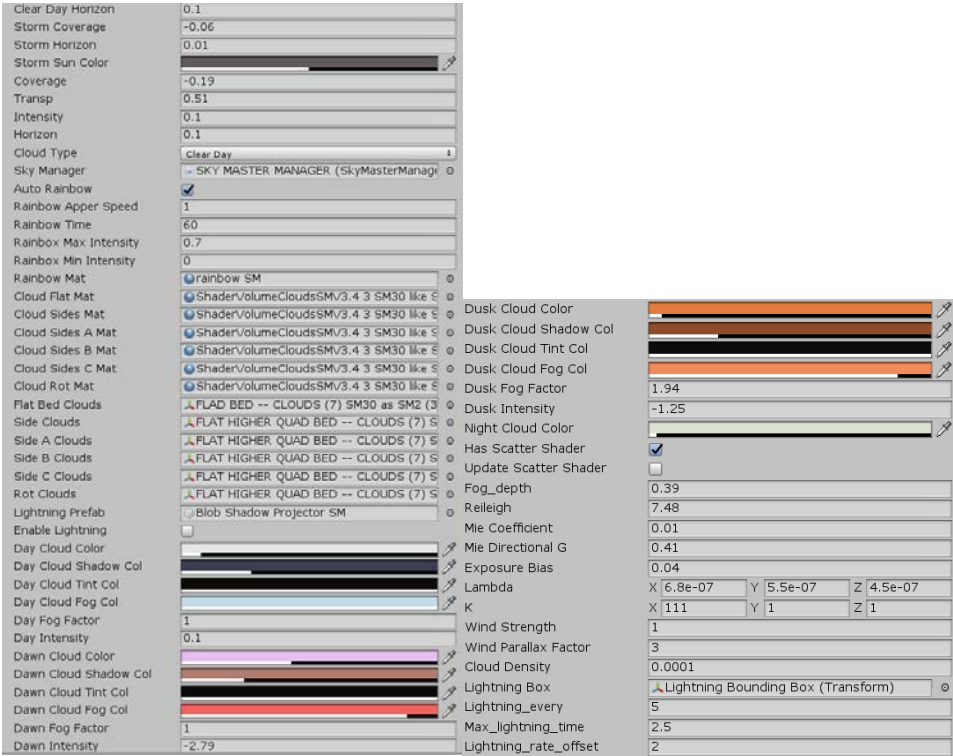
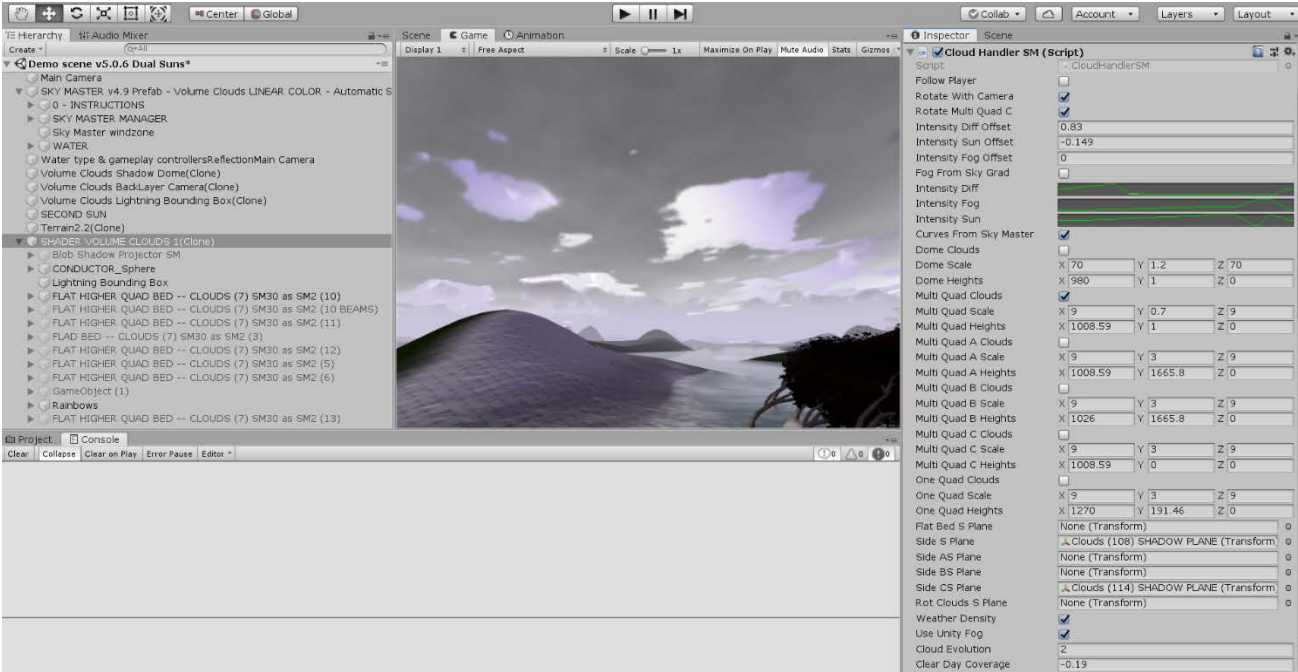
The Mesh based volumetric cloud system creates a volumetric effect using multiple quad or semi spherical dome layers stacked on top of each other with a small distance between them.

To setup the system in the scene, press the **“Add Shader based volume clouds”** button in the Configurator.

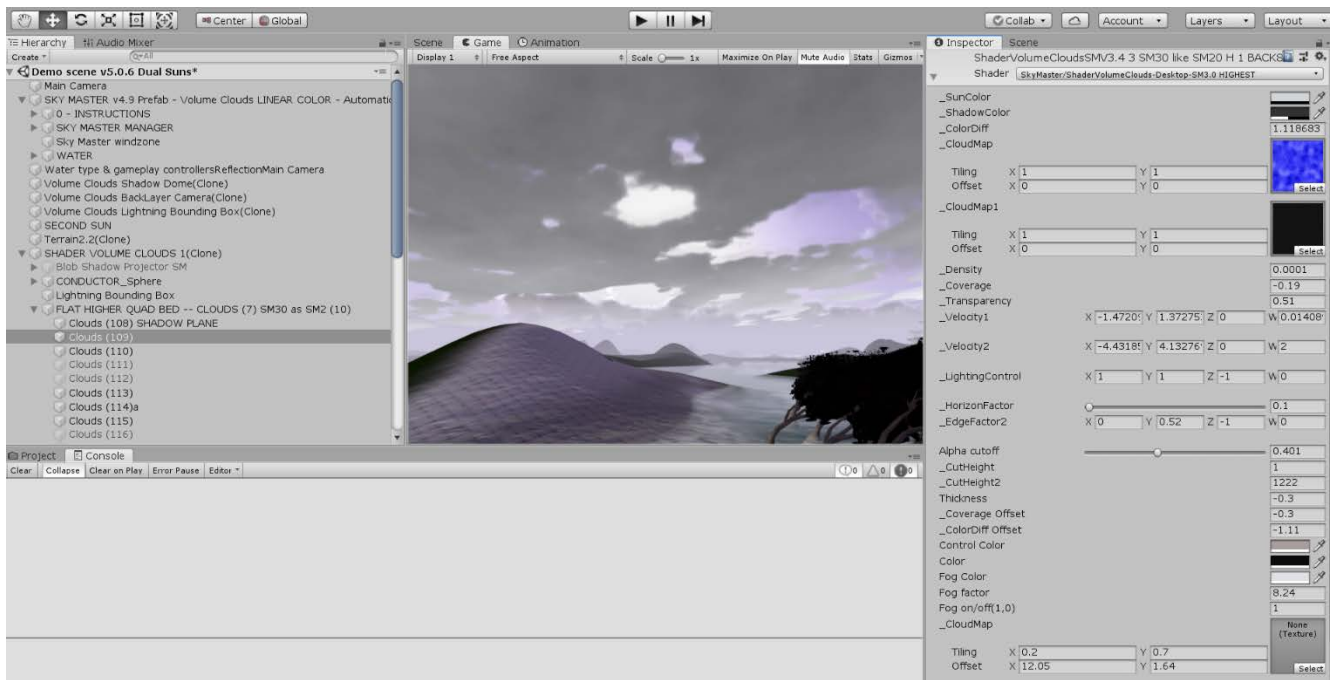


This will instantiate a ready-made setup, where the cloud layers are grouped per type and parented to a gameobject (**“SHADER VOLUME CLOUDS 1”**) that holds the cloud controller script named **“CloudHandlerSM”**. The script references the various cloud layer groups corresponding the different cloud types (**“Flat Bed, Side A-B-C and Rot clouds”**), their materials (**“Cloud Flat – Side A-B-C-Rot Mat”**) and their shadow layers (**“Flat – Side A-B-C- Rot S Plane”**).

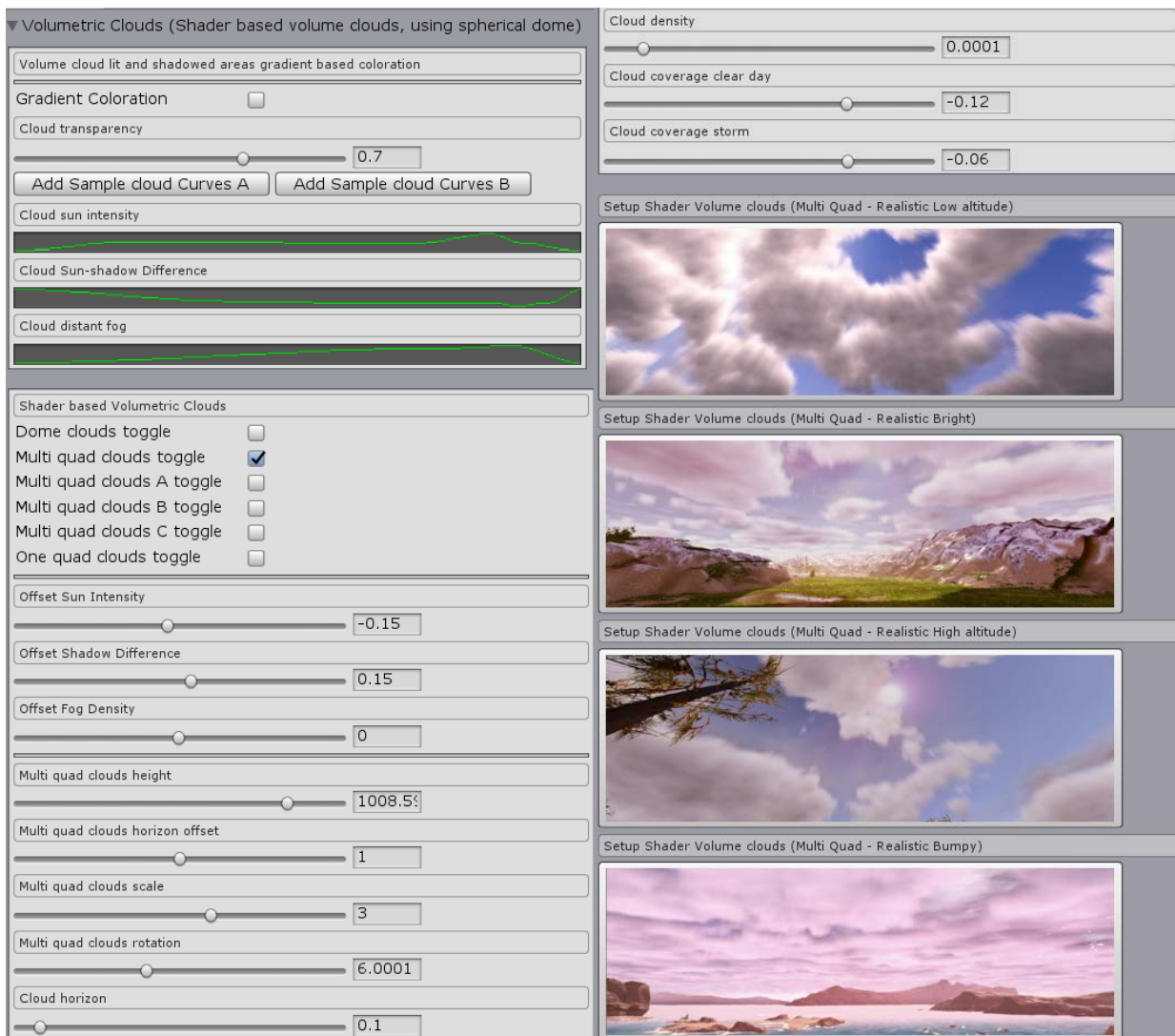
Each cloud type can be configured separately so can have multiple presets to use directly, by enabling the **“Dome Clouds”, “Multi Quad Clouds A-B-C”** and **“One Quad Clouds”** checkboxes and setting their scale and height properties. The cloud selection is also configurable through the Configurator, without need to enable the checkboxes in the script directly.



Each of the cloud planes is assigned the volumetric cloud shader, as shown in the following image.

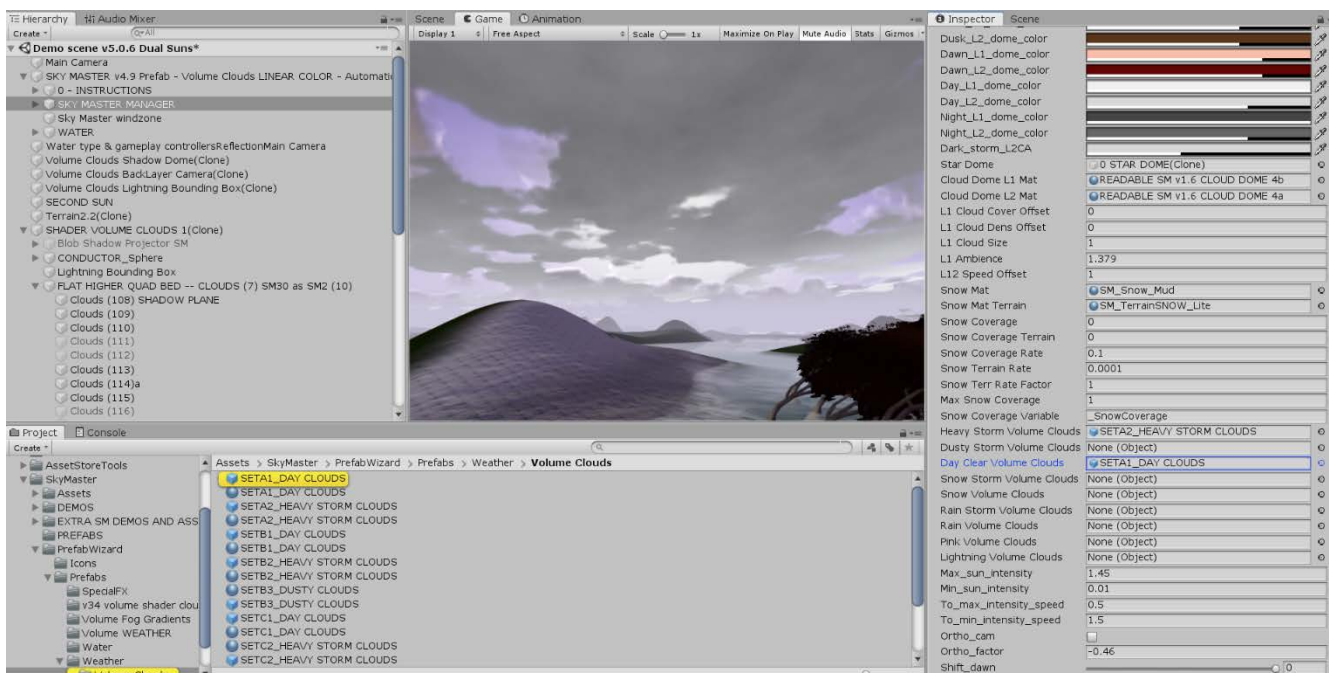
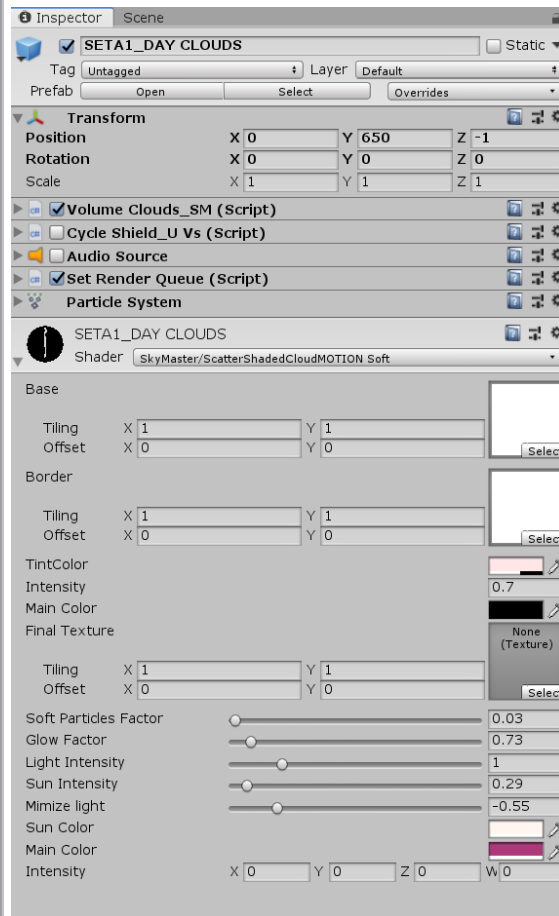


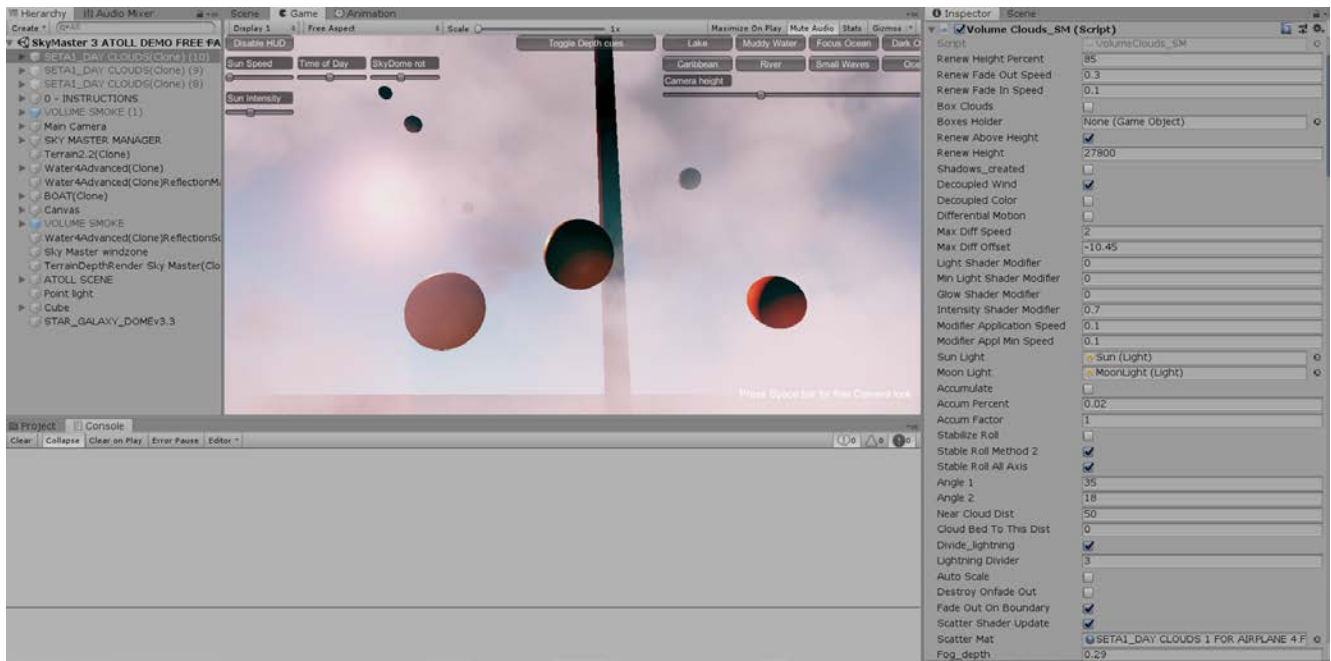
The main properties and cloud selection of the system can be configured in the Configurator in Clouds section as shown in the following image.



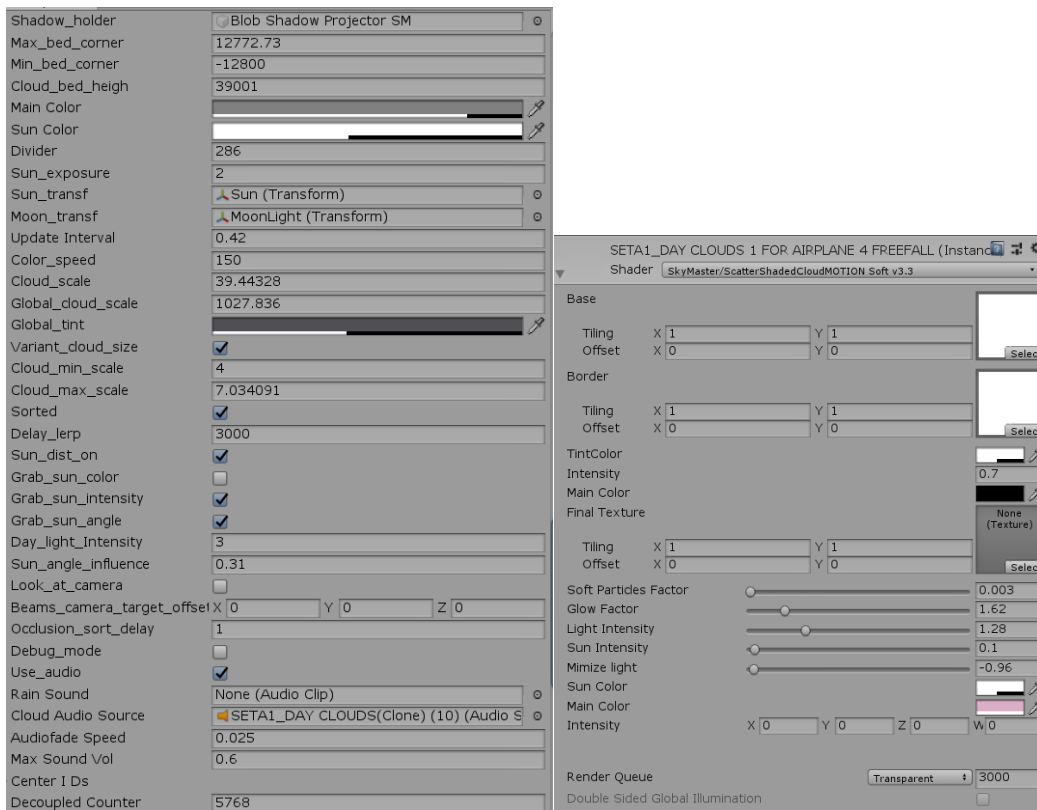
5d. Particle Volume Clouds

The particle based volumetric system uses Shuriken particles with a volumetric shader to create a volume effect. The system allows to shape clouds as needed, so is useful when a very specific shaping of the clouds is required.



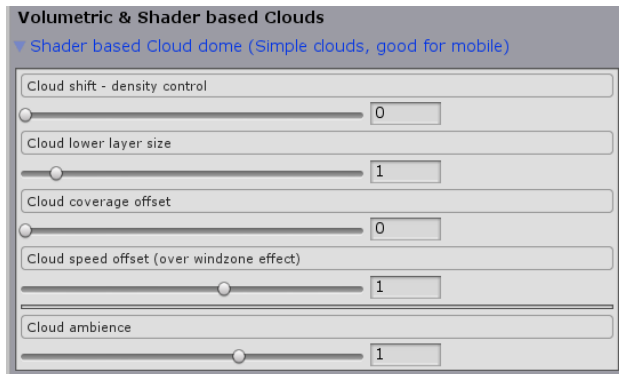


Relleigh	0.005670947	Dawn_sun_col	
Mie Coefficient	0.0004	Dusk_base_col	
Mie Directional G	0.1	Dusk_sun_col	
Exposure Bias	0.11	Night_base_col	
Lambda	X 6.8e-07 Y 5.5e-07 Z 4.5e-07	Night_moon_col	
K	X 0.9 Y 0.5 Z 0.5	Add_shadows	<input checked="" type="checkbox"/>
Vertical Form	<input type="checkbox"/>	Use 2D Check	<input type="checkbox"/>
Y Scale Diff	420.57	Max_divider	14
Max Cloud Dist	X 2.11 Y 254.2 Z -1.4	Cloud_spread_delay	0
Yspread	616.92	Method	3
Height Tint		Turbulent	<input type="checkbox"/>
Smooth In	<input type="checkbox"/>	Flatten_below	<input checked="" type="checkbox"/>
Smooth Out	<input type="checkbox"/>	Shaft Centers	
Smooth In Speed	1.108503	Sun Shafts	
Smoothout Speed	0	Beam_Occlusion	<input type="checkbox"/>
Smooth In Rate	1.05	Smooth_mat_trans	<input type="checkbox"/>
Smooth Out Rate	0.05	Appear_Beam_speed	1
Max_fade_speed	1.1	Disappear_Beam_speed	1
Max_smooth_out_time	1	Shaft Scale	
Fade_in_time	2.5	Beam_length	3300
Current_fadein_time	0	Scale_beams	<input type="checkbox"/>
Enable LOD	<input type="checkbox"/>	Scale_on_collision	<input type="checkbox"/>
LOD_send_height	-10000	Offset_col_factor	0.3
Partides LO Ded		Diminish_beams	<input type="checkbox"/>
Lod Max Y Diff	360	Override_sun	<input checked="" type="checkbox"/>
Lod Max H Diff	2100	Cut_height	-20000
LOD Fade In Speed	15	Restore_on_bound	<input type="checkbox"/>
LOD Fade Out Speed	64	Disable_on_bound	<input type="checkbox"/>
LOD Min Dist	150	Bound	9000
Cloud Waves	<input type="checkbox"/>	Destroy_on_end	<input checked="" type="checkbox"/>
Wave Freq	0.3	Clone_on_end	<input checked="" type="checkbox"/>
Moon_light	<input checked="" type="checkbox"/>	Cloned	<input type="checkbox"/>
Moon_light_color		Centers	
Moon_dark_color		Centers_Init	
Override_init_color	<input type="checkbox"/>	Speed	3
Override_color		Multiplier	1
Day_cycle	<input checked="" type="checkbox"/>	Wind	X 0 Y 53.3504 Z 0
Sky Manager	SKY MASTER MANAGER (SkyMasterManager)	Get_wind_direction	<input type="checkbox"/>
Day_base_col		Wind_holder	None (Game Object)
Day_sun_col		Rain_holder	LIGHT RAIN/voICI
Dawn_base_col		Lightning_holder	Quad VoICI Lightning



5e. Simple clouds

The system includes a simple cloud shader for use with older mobile platforms or as extra cloud layer with the rest of true volumetric cloud types described above. These clouds are by default included when the sky system is created, in the SKY MASTER MANAGER gameobject (**“DOME CLOUD L1-2”** game objects). Their main properties can be controlled by the Configurator section shown in the following image.



6. Volumetric fog

The system includes a gradient colored volumetric height fog module that can be used to give depth and occlusion to the scene and various background effects. It supports atmospheric scattering on the distance and depending on the sun positioning.

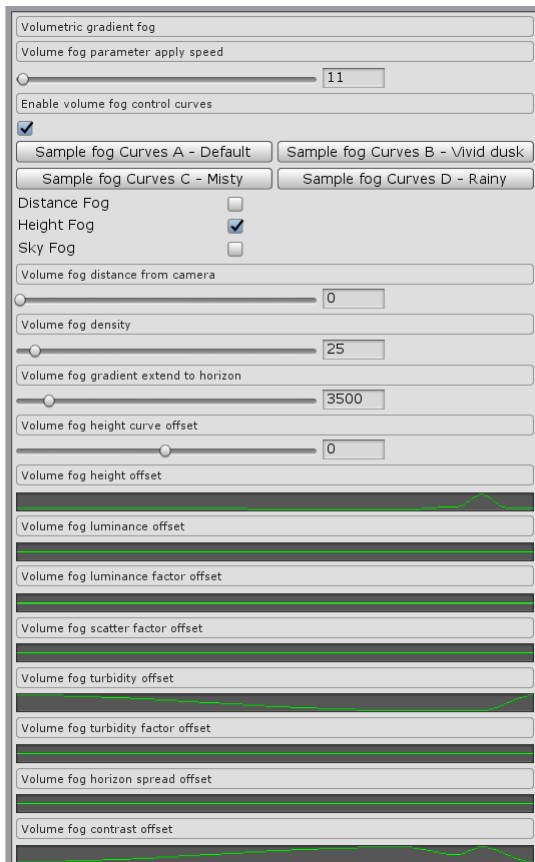
To setup the module on the main camera, press the “Add volumetric fog” button.

NOTE: When script is used as standalone or copied from another scene camera, the Sun of the scene must be dragged in the relevant sun slot of the script on camera.



The fog is controlled by two methods. The default method is by presets, that are hardcoded in the “SeasonalTerrainSKYMASTER” script, which is setup through the terrain setup process or when creating a new Sky Master setup in latest versions of the system, the script in latter case is found in the same gameobject as the SkyMaster configurator.

The **latest and preferable method** is though a sliders and curves system, which is activate when the “Enable volume fog control curves” checkbox is enabled. Then the settings in the section shown in the image below will directly control the main fog parameters on the script on camera (“GlobalFogSkyMaster” script).

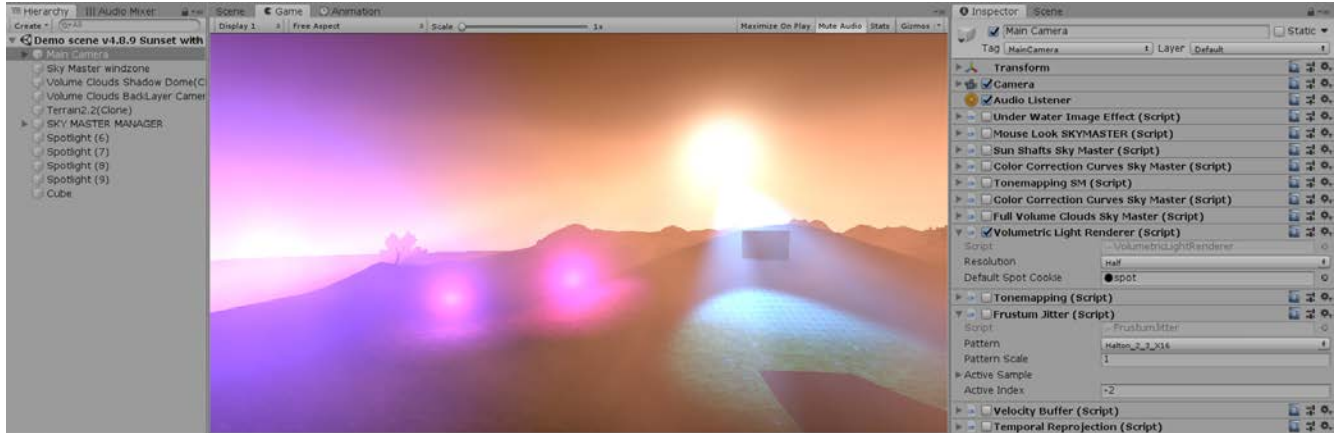


7. Volumetric lighting

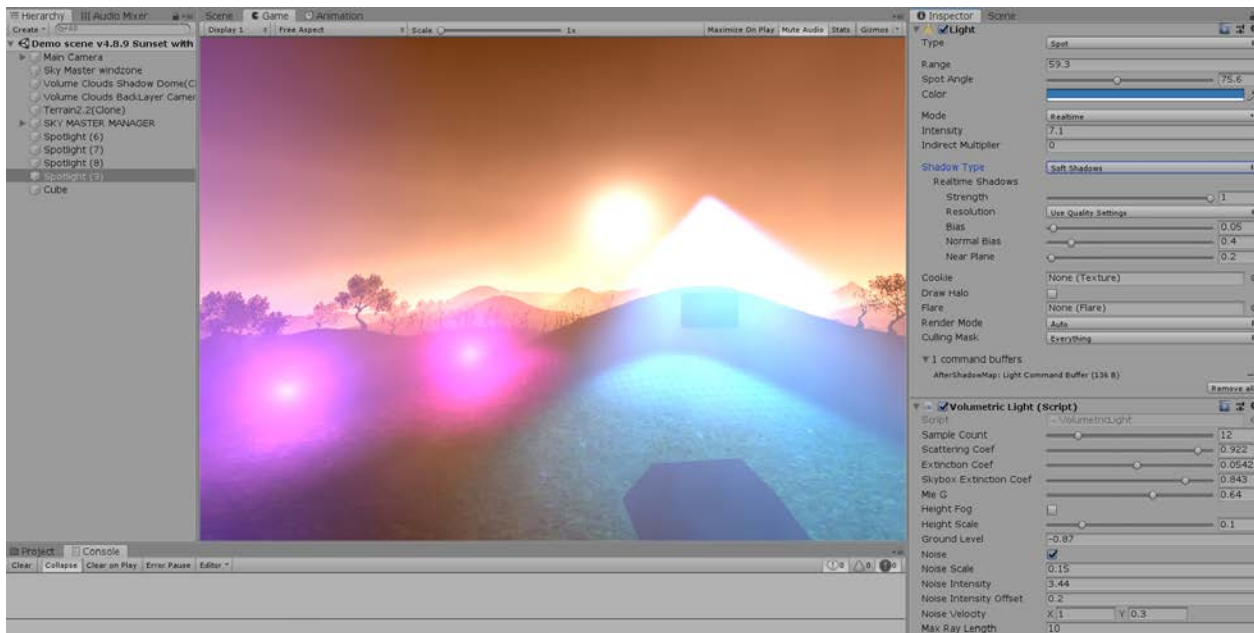
The system offers two volumetric lighting modules, one is solely for the sun light and the latest one which is in Alpha offers volumes in all light types (Spot, Point and Directional).

7a. Full volumetric lighting

In order to setup the full volumetric lighting system, add the **“VolumetricLightRenderer”** script on the main camera of the game .



For every light that requires to be rendered in the fog volume, insert the **“VolumetricLight”** component and setup the various parameters for volume quality. To get shadows on the volume, the light must have soft or hard shadows enabled.

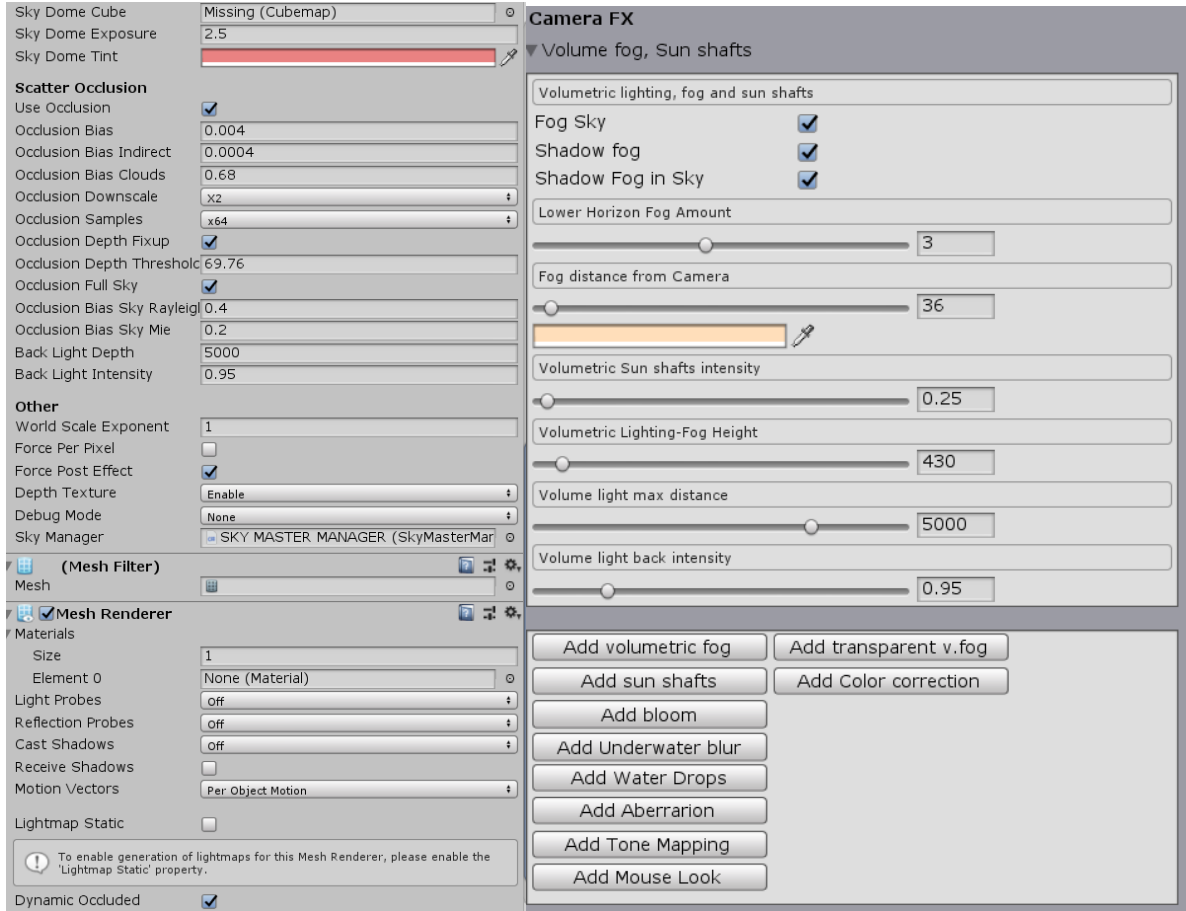
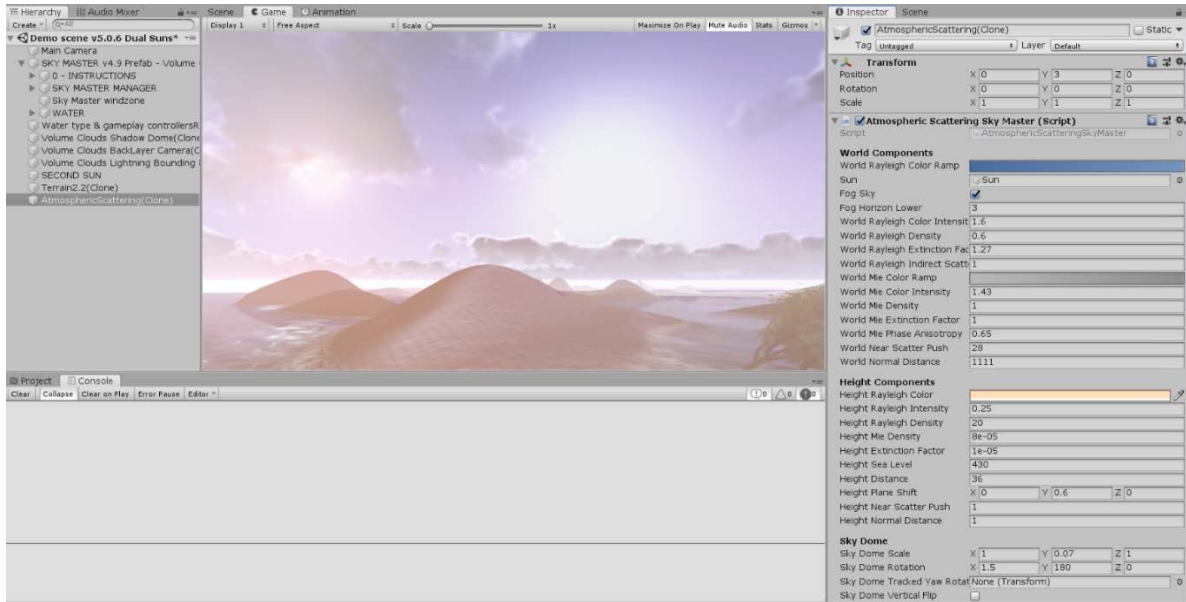
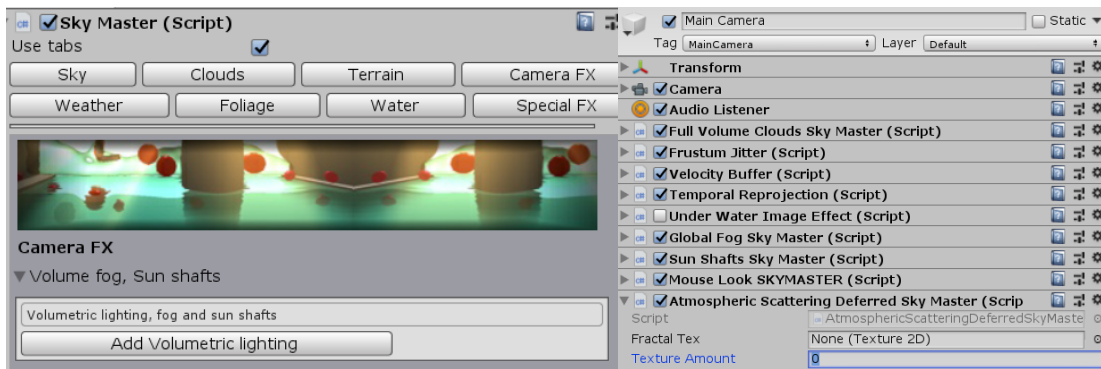


7b. Sun volumetric lighting

The volumetric sun light system allows to have full volumetric lighting from the Sun. To setup the system, press the **“Add Volumetric Lighting”** button to add the system to the camera (**“AtmosphericScatteringDeferredSkyMaster”** script).

The system also adds a **“AtmosphericScattering”** gameobject in the scene, that has the component named **“AtmosphericScatteringSkymaster”** and a mesh required for the effect rendering.

NOTE: If the effect seems to not update during gameplay, erase the Mesh Filter and Mesh Renderer of the **“AtmosphericScattering”** gameobject and they will be recreated for the run time.



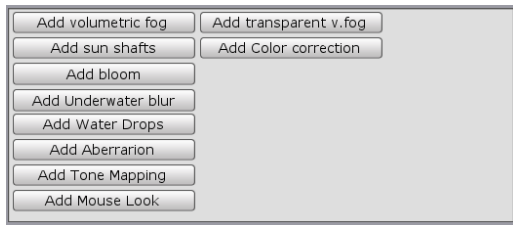
The main properties of the system can be controlled through the Configurator as shown in the images above, right panel.

7c. Simple Sun volumetric light

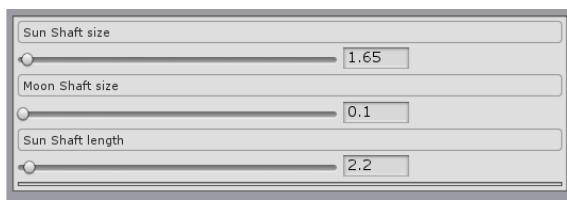
The system includes a simple volumetric light sun shafts solution, for use with mobile and as additional rays in the full volumetric lightning systems.

To setup the system on the main camera press the **“Add sun shafts”** button.

NOTE: If the script is used as standalone or copied from another scene camera, the main scene sun light must be referenced in the relevant slot of the script on the main camera.

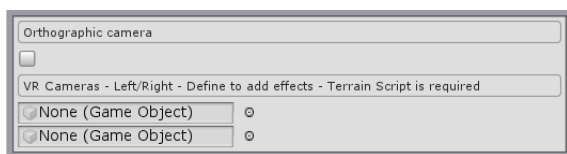


The main properties of the sun shafts can be set in the section shown in the following image, in the Configurator.



8. 2D Game and VR setup

The system supports 2D sky, when using an orthographic camera, by enabling the **“Orthographic camera”** checkbox.



For VR setup reference, use the demos named **“Demo scene v5.0.5 - 3D NOISE CLOUDS with WATER VR”** and **“Demo scene v5.0.5 - 3D NOISE CLOUDS without WATER VR”**.

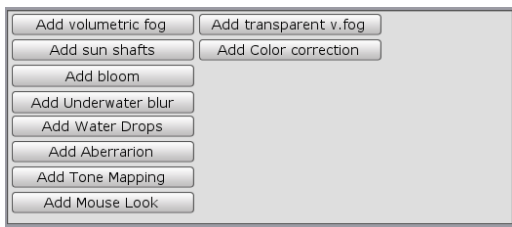
The demos showcase use of the volumetric clouds, background items rendering – blending with clouds and water reflections in VR.

9. Extra image effects

The asset provides a number of image effects that can be used to enhance the overall look, including

- Bloom for vivid specular reflections
- Blur (for underwater or general use)
- Water drops which implements a quad in front of the camera that emulates water dripping on rain or when leaving water and freeze effects
- Chromatic Aberration effect
- Tone Mapping for regulating the image contrast
- Color correction for color regulation per RGB channel

The effects will be setup in the Main Camera when pressing the corresponding button in the menu shown in the following image.



10. Global Illumination

The system has two different real time GI methods, the GI Proxy for emulating GI using shadow-less point lights, which works with every platform and the SEGI system, which is used for more detailed and global real time GI implementation.

10a. GI Proxy

The Global Illumination Proxy (GI Proxy) system emulates GI using shadow-less point lights and works with every platform since it uses the standard Unity lights and no image effects. A detailed usage guide is included in the GI Proxy folder.

10b. SEGI

The SEGI system is derived from the work found in [SEGI Github](#) (MIT License), with custom modifications that enhance performance of the core system to double or more frame rate so can be used in realistic game scenarios.

Refer to the **"User Guide.pdf"** inside the **"ARTnGame GI PROXY"** -> **"SEGI"** folder, for usage instructions. Use the **"Update every"** variable in the **"SEGI"** script on camera, to enhance the system performance, forcing the GI calculations to happen at longer intervals.

11. Extra effects and modules

Atomic Bomb



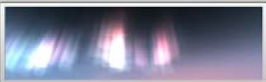
Asteroid field



Freeze effect - Ice decals



Aurora



Chain Lightning



Sand storm



Volcano

