

# Sky Master ULTIMATE HDRP

For quick start on the main system, please refer to my tutorial videos in the following links (standard pipeline):

All tutorial videos:

[https://www.youtube.com/watch?v=EveE2IcXGeI&list=PLJQvoQM6t9Ge2ehO4N1kNq3jvHmVst\\_el](https://www.youtube.com/watch?v=EveE2IcXGeI&list=PLJQvoQM6t9Ge2ehO4N1kNq3jvHmVst_el)

Script reference and guides- manual can be downloaded from the following links:

Manual:

[https://drive.google.com/drive/folders/1o\\_DusjNj8O77vMY4aEbZFp1Z\\_15\\_3MV8](https://drive.google.com/drive/folders/1o_DusjNj8O77vMY4aEbZFp1Z_15_3MV8)

Script reference:

<https://drive.google.com/open?id=1tlpFnWzFOErDIVDLiZc99rK6qABhKPy9>

For any questions please contact me in my discord channel:

<https://discord.gg/X6fX6J5>

Or in my email:

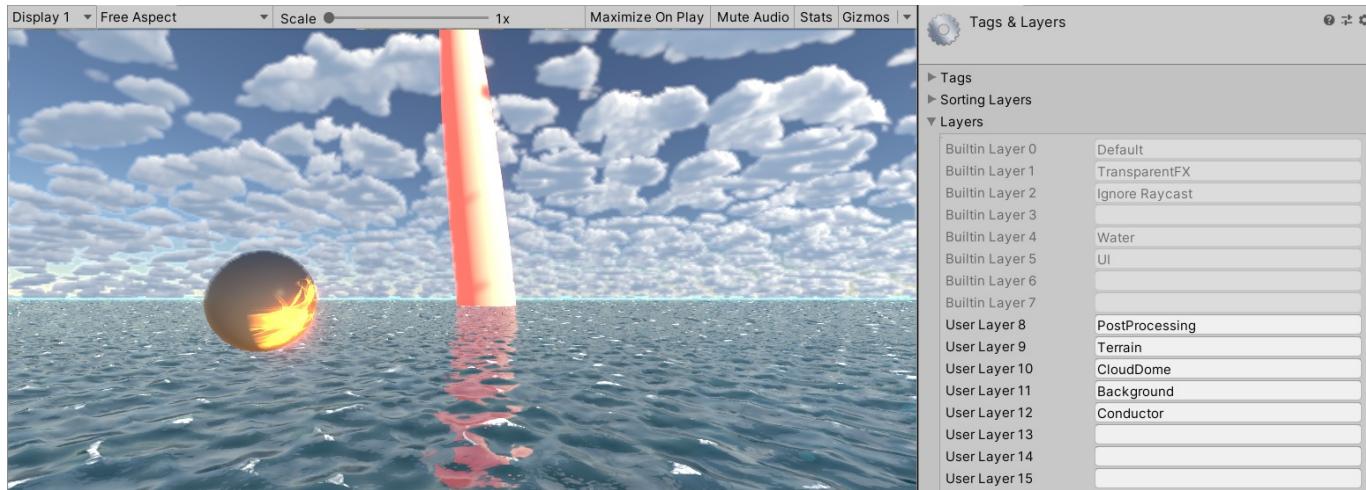
[artengames@gmail.com](mailto:artengames@gmail.com)

---

## HDRP specific setup

---

### 1. Add layers to the project, named as shown in the photo below



### 2. Insert the HDRP Beta package to the project

### 3. Assign the HDRP Asset ("HDRRenderPipelineAsset\_SKY\_MASTER") in Project Settings -> Graphics in "Scriptable Render Pipeline Settings" slot and in "Quality" section in Rendering slot.

### 4. Drag any of the prefabs in the "PREFABS" folder root in the scene after deleting the current directional light and camera. Then the camera can be replaced by any other, as long as it is tagged as "MainCamera" and the one in the prefab is deleted. Also all scripts of the prefabs camera should be copied to the other camera before deleting it.

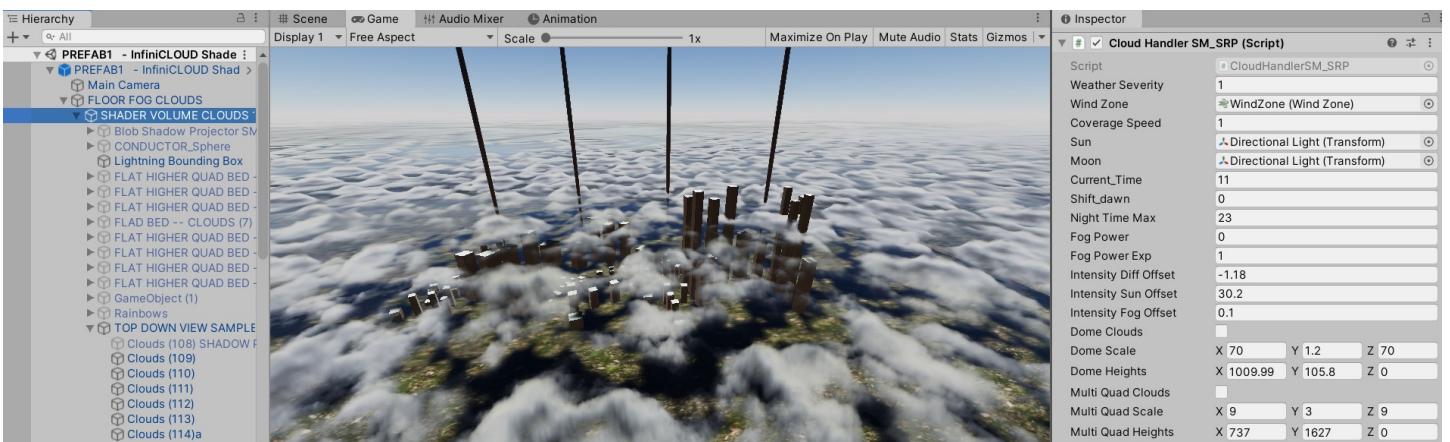
**5. "SkyMaster" script can be used to configure the Time of Day and sun position, directly grabbed from a prefab or by inserting the "SkyMaster" script in an empty gameobject.**



**6. The HDRP system includes a volumetric clouds system with sun shafts ("connectSunToFullVolumeCloudsHDRP") which supports two image effect based cloud types, selected using the "Cloud choice" slider. Use "0" to select the background volumetric clouds or "1" for the full volume clouds that support fly through.**



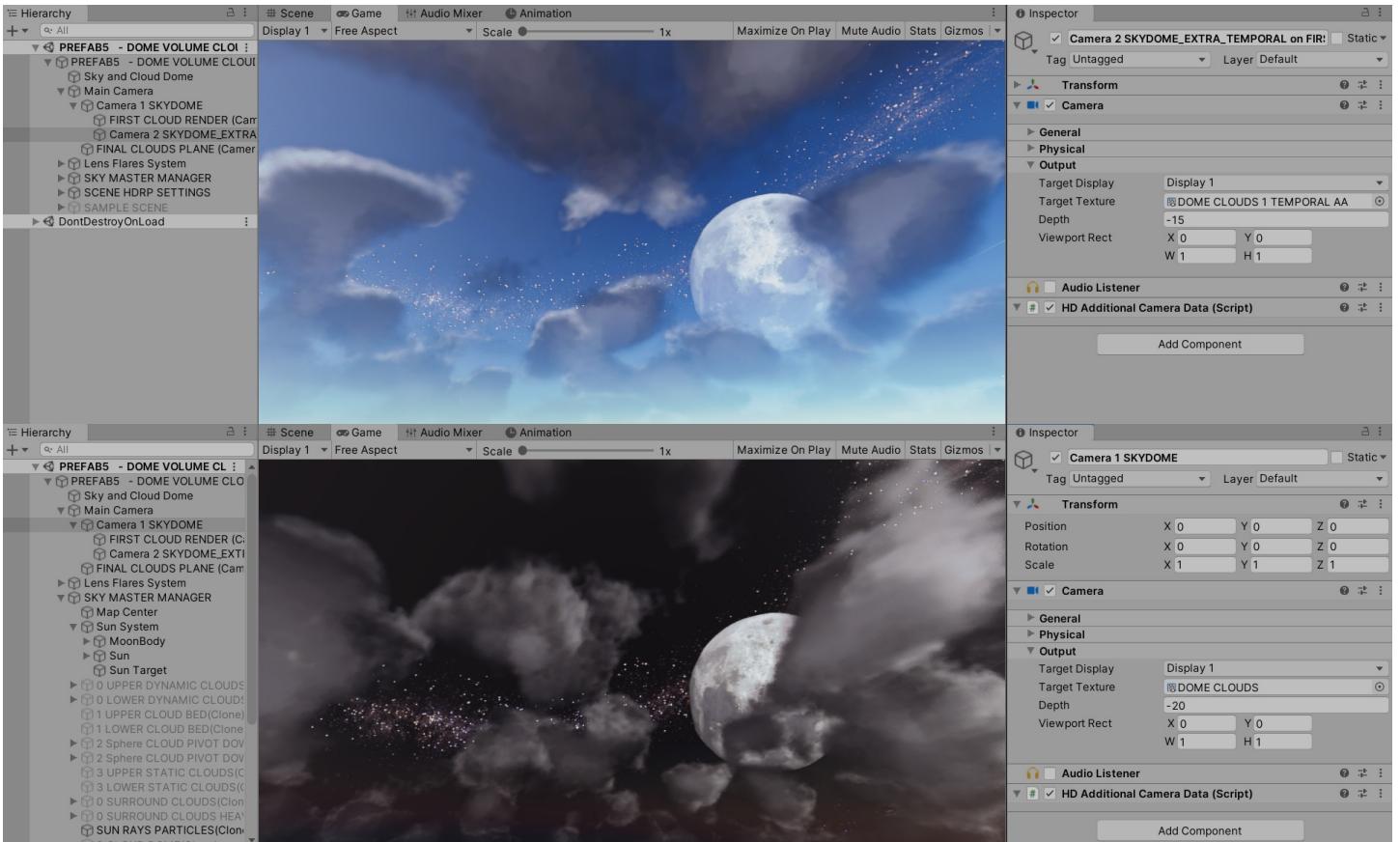
A 3rd cloud system is included which renders volume clouds using a shader applied to stacked quads in the scene, this is pure shader based (InfiniCLOUD) and does not require any image effects or pipeline render features. The controller for this module is "CloudHandlerSM\_SR" script.



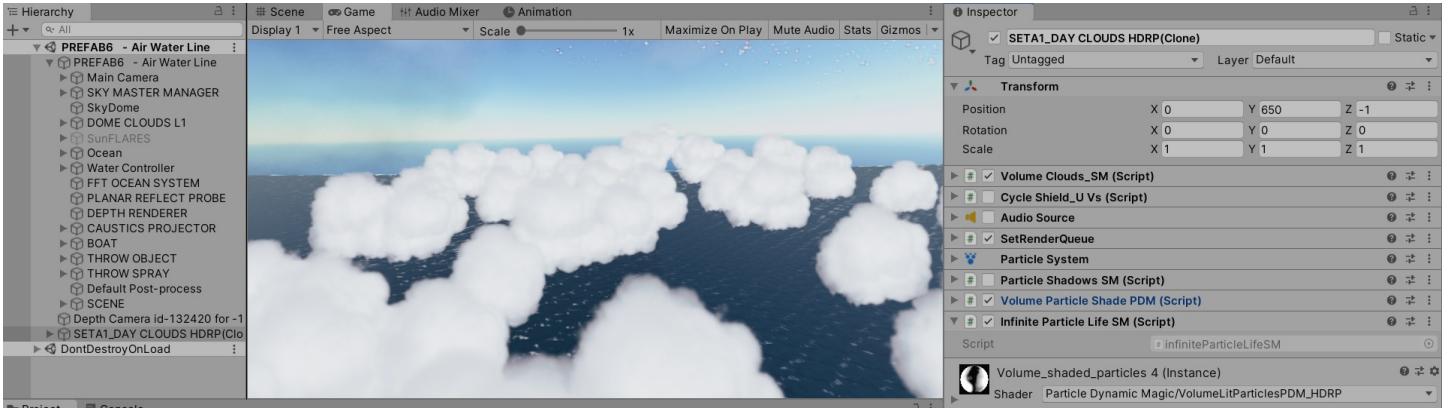
The 4th Cloud type is same as the "0" choice clouds in the "connectSunToFullVolumeCloudsHDRP" system, but is implemented in shader and rendered in a dome in the scene, so no image effects are required. This may be useful in some situations where image effect may not be desired or compatible with other scene features. Also with this system is possible to do cheap screen space reflections of clouds in water and use HDRP own Temporal AA to reduce cloud noise when sampling is low for increased performance. The shader also integrates a sky with atmospheric scattering, a moon and a galaxy for a complete sky rendering without using any extra geometry or shaders.



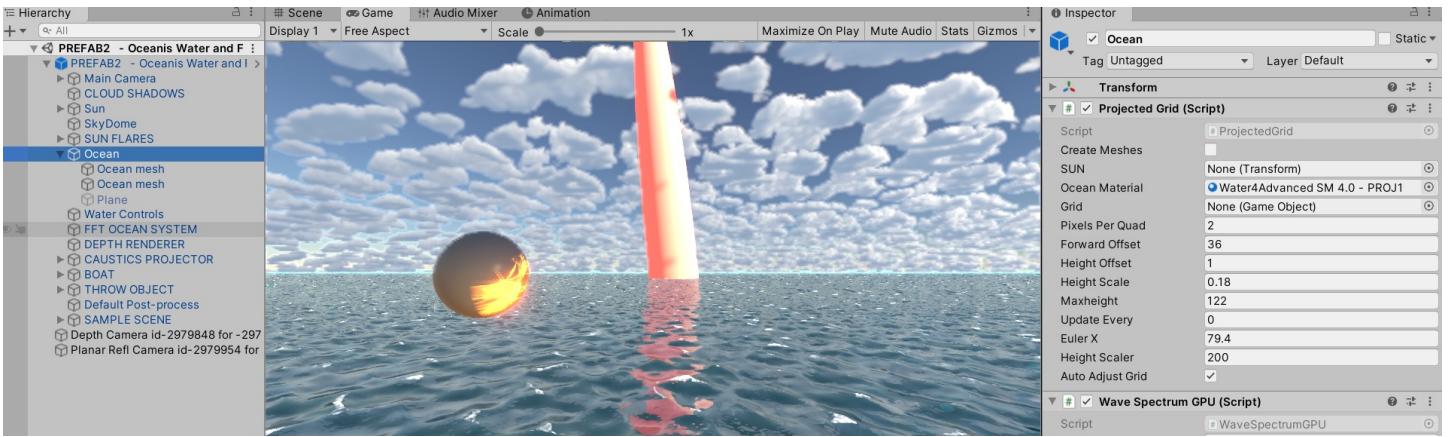
The dome is in the Clouds and Sky Dome object in the demo scenes, and is a sphere with the volume clouds shader material and a "Sky" script that controls the material and the sky properties for atmospheric scattering. In order to use the Temporal only in clouds, a system of 2 or 3 (for extra AA) cameras is used, one renders the original dome with the clouds only using Temporal AA (CloudDome layer) and passes the rendertexture to a quad, which is directly rendered from the main camera or in 3 cameras case rendered from a 2ond camera using extra aTemporal AA (using the "Background" layer) and then applied to another quad that is finally rendered from the main camera.



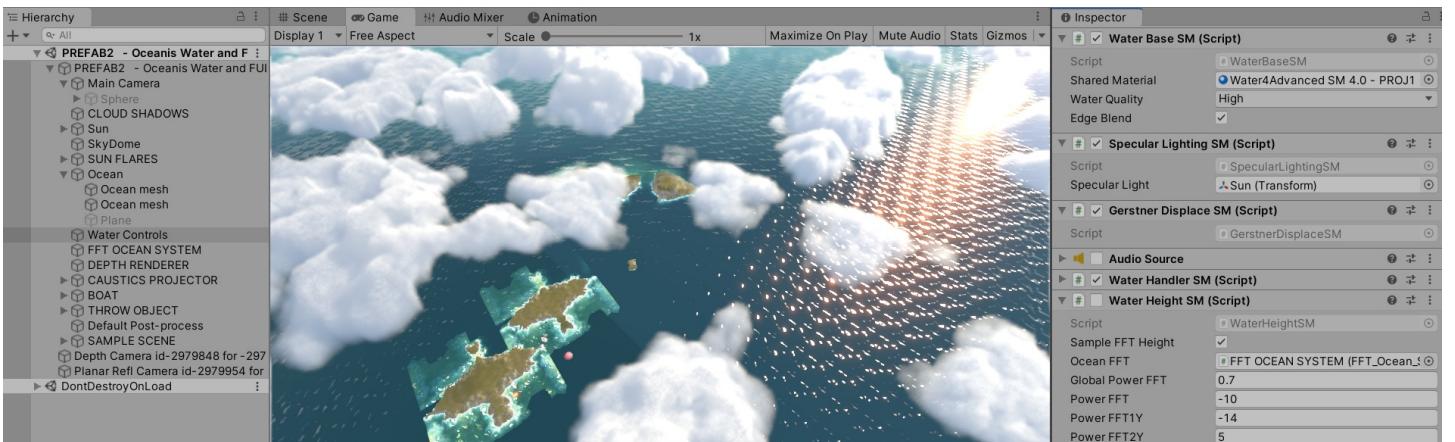
A volume lit particle system is also included in the system, for use with Shuriken. This system can be used for moving clouds with specific formations and for effects like smoke or volcano.



The system also includes an advanced “Ocean and Water” module, the base scripts is the “Projected Grid” and “Wave SpectrumGPU” in the “Ocean” object and the “FFT\_Ocean\_SM3” script in the “FFT OCEAN SYSTEM” object.



The “Water Controls” object holds the Base, Specular, Reflection, Gernstner, Water controls (“WaterHandlerSM”) and Dynamics controls (“WaterHeightSM”) for the water.



The water system also has a special air - water separation system that can enable image effects only below the water. The effects include sun shafts and water flow, refraction line in the water - air border adjustable in width. The effect is implemented in the HDRP image effect "SunShaftsSM\_HDRP" which must be enabled in a volume module in the scene and is then controlled by the script "PassMaskToEffectSM" which must be attached to the gameobject that has the relevant Volume component.



## 7. Helper scripts

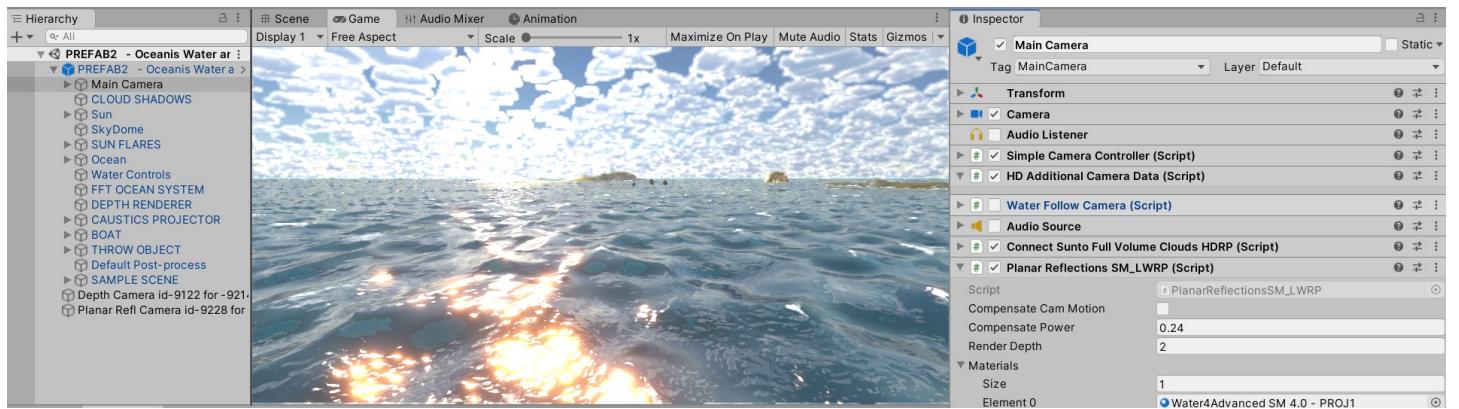
The air-water separation system uses a black thin line enabled in the water in its shader, to control this line a helper script is used on camera named "ControlReflectDepthFX\_HDRP", in Black Line section and Camera Near Clip Plane depending on camera near clip plane.

The same script is also used to control properties of reflection (power, tiling which may vary depending on whether a Reflect Probe is used or the custom reflect script of Sky Master HDRP), also is used to reset the water waves main power when a scene starts, because the water system has a method to reduce the wave power as camera goes way above ground and if play mode exits at that point water main waves are zeroed.

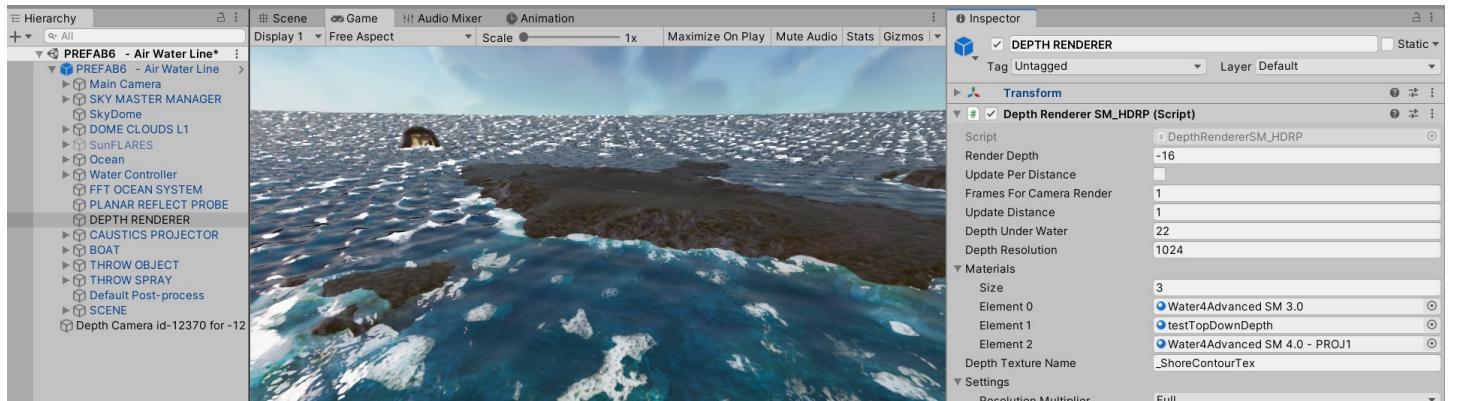
The Depth Pyramid Scale is a set of variables controlling the camera depth texture scaling, in some cases this is required to be (1, 0.67) and in other cases must be automatically adjusted using the "AutoDepthScaleUpdate" checkbox.

The OffsetScale variable is used to control the Top Down Depth Camera renderer texture, which is used to regulate the water in shores.

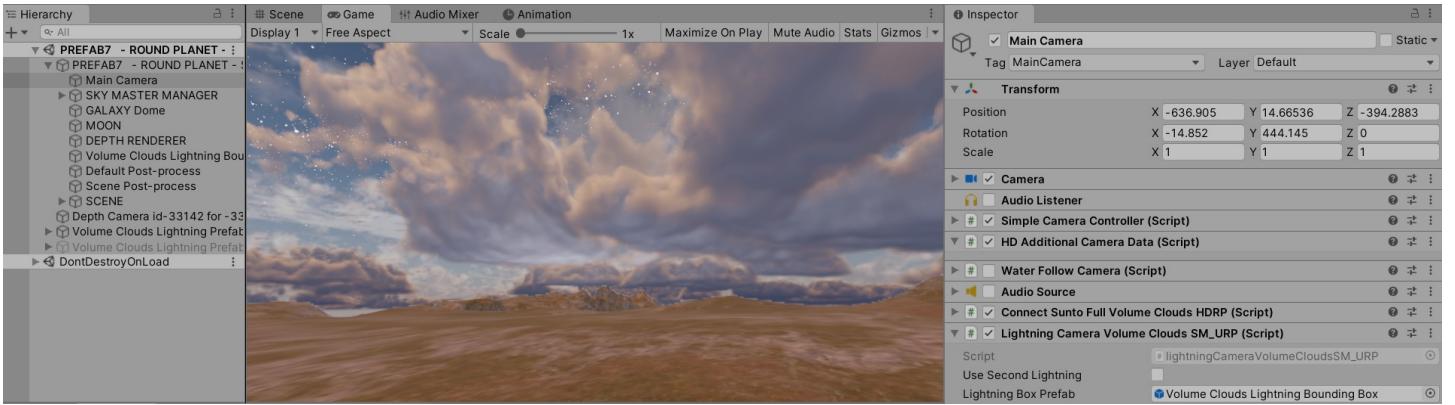
**REFLECTION SCRIPT:** The system uses a custom reflection script on camera "PlanarReflectionsSM\_LWRP" to render a reflection texture of the scene, including the custom image effects like volumetric clouds. Use this script when need to include the clouds in the reflection. Otherwise use a Planar Reflection Probe, with the "ControlPlanarReflectProbeSM" script attached to the probe gameobject.



**SHORE LINES:** In order to regulate the shore lines, the "DepthRendererSM\_HDRP" script is used on camera, which creates a top down depth rendering orthographic camera, which is rendering the terrain by the selected layer and inserts the texture to all needed materials as referenced in the "Materials" section of the script. The variable "DepthWorldScaleMultiplier" defines the world units of the top down depth rendertexture, used for the shorelines handling. The parameter depends on the game resolution, for 1280x720, set it to 2.8. If other resolution is used, must be manually adjusted so the depth matches the land. To debug set "ShorePower" vector Z value in the water shader to 20 so water overlaps the land for visual feedback of the depth texture fit on the landscape.



**LIGHTNING:** The lighting is handled by the "LightningCameraVolumeCloudsSM\_UPR" script on the main camera. The script uses a bounding box (a simple box object with collider and mesh renderer disabled) to randomly spawn the lightning prefabs that create the lightning in the scene inside its collider volume. The system also passes the lightning light position to the volumetric clouds connect script in order to create the light of the lightning on the clouds volume.



**CLOUD SHADOWS:** The system uses a separate shader "HDRP SHADOWS SM CLOUDS FULL VOLUME" to render clouds shadows. The shader has three modes selected by the "Shadow Mode" variable in the material, one is for the shader based clouds ("0"), the other for the background clouds ("1") and the last for the fly through clouds ("2"). The parameters must be manually replicated for the relevant cloud type. The automation of passing the parameters based on the cloud types is currently work in progress for the next HDRP Beta version.



## 8. Quick Tips:

**NOTE 1:** You may use the WeatherRandomizerSKYMASTER and TimerSKYMASTER scripts, attach them to the skymaster object and trigger random weather effects or set precision time. Those can be also used as base templates for scripting the system, they showcase how to change weather and time of day by setting up the proper variables.

**NOTE 2:** Add the "Conductor" Tag in the project and in any item that would like to receive lightning in a storm. Add the "Flamer" and "Flammable" tags to use the special FX like propagating fire and ice.

## 9. Known issues in HDRP Beta:

**ISSUE 1:** If notice any artifacts or scaling in screen texture after loading a HDRP demo scene, press play button to see if fixes the issue. If not, create a new scene and then reload the desire scene and press play to address the issue.

**ISSUE 2:** The system may use multiple cameras besides the main, e.g. one camera looking down to render depth for shore lines and one for planar reflections. The order of rendering those cameras may severely affect the result of their use, thus is advised to keep the order as suggested in the demo scenes.

**ISSUE 3:** In HDRP 10.2, the shadows require an upgrade to work properly.

If HDRP 10.2 or newer is used, the shadows shader must be upgraded using the following package  
"SkyMaster ULTIMATE HDRP Beta 4\_\_HDRP\_10\_2\_UPGRADE\_for\_SHADOWS"

The package is found in the following folder:

Assets -> SkyMaster -> HDRP SHADOWS -> 0 HDRP\_10\_2 UPDATE

Move to the folder and double click on the package to setup the HDRP 10.2 shadows.