Oceanis URP Pro 2024 Water Setup Guide

Overview

The **Oceanis URP Pro 2024 Water** system is a water framework for endless Ocean and water bodies creation in URP. The system is compatible with **Sky Master ULTIMATE URP** version, and provides advanced features like dynamic water foam system, multi scale wave simulation, endless projected grid ocean, volumetric lighting and fog for underwater, water-air separation for underwater effects just under water surface. Also the system can work together with **InfiniRIVER** system, using the river system 2D liquid simulation for dynamic foam for shorelines or smaller bodies of water.

For quick start on the main system, please refer to the tutorial video playlist in the following link:
https://youtu.be/uRD08e8e-HU

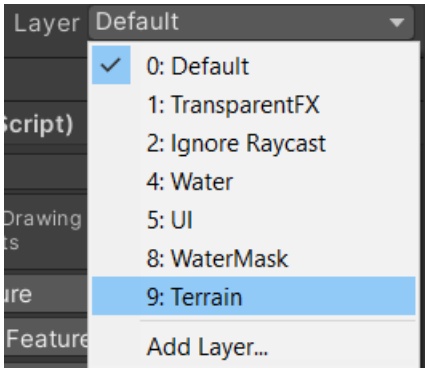
For any questions please contact me in my discord channel:
https://discord.gg/X6fX6J5

Or in my email:
artengames@gmail.com

Oceanis Setup steps

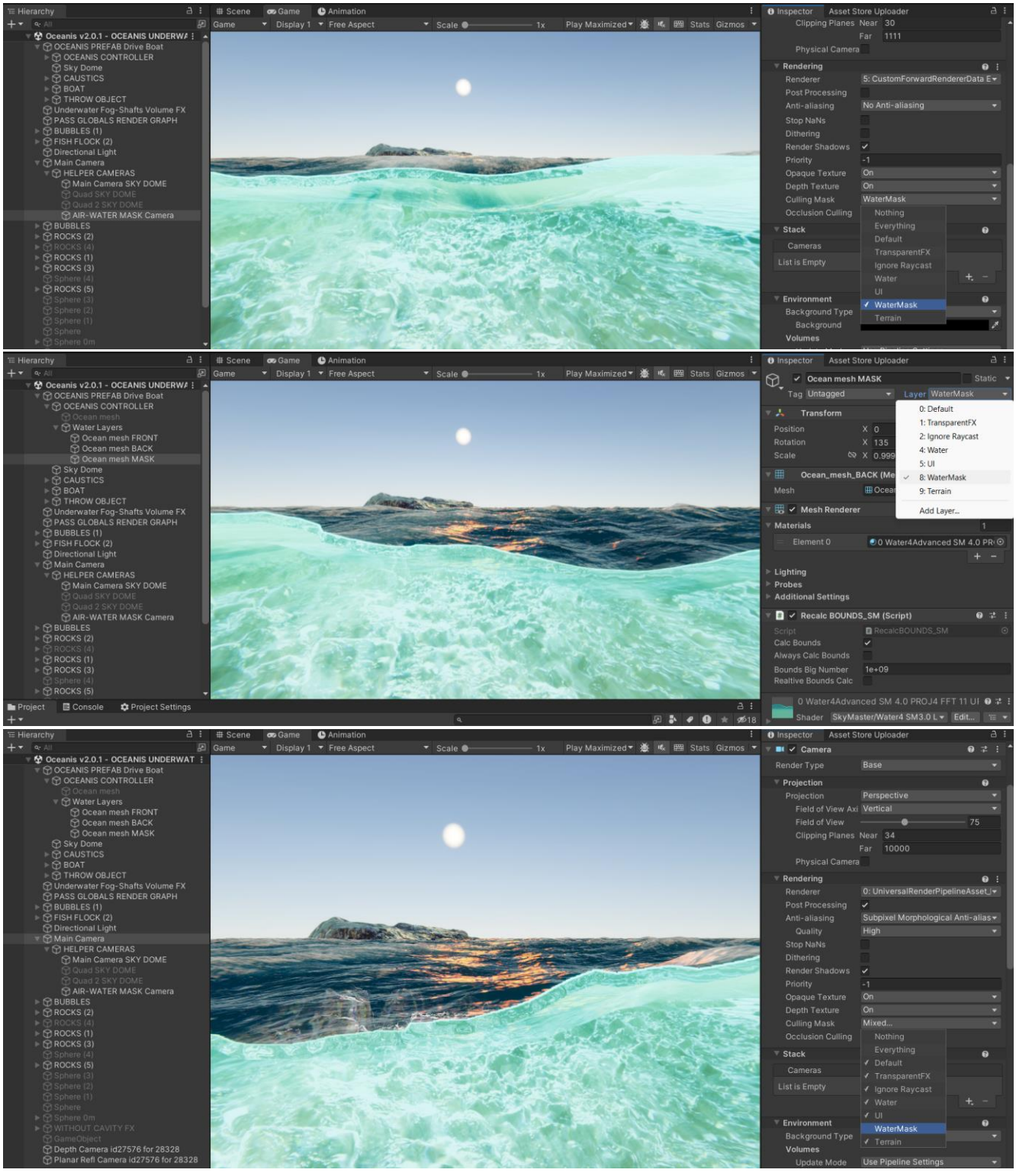
Before the setup a few layers and tag should be added to handle the various system functionalities.

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



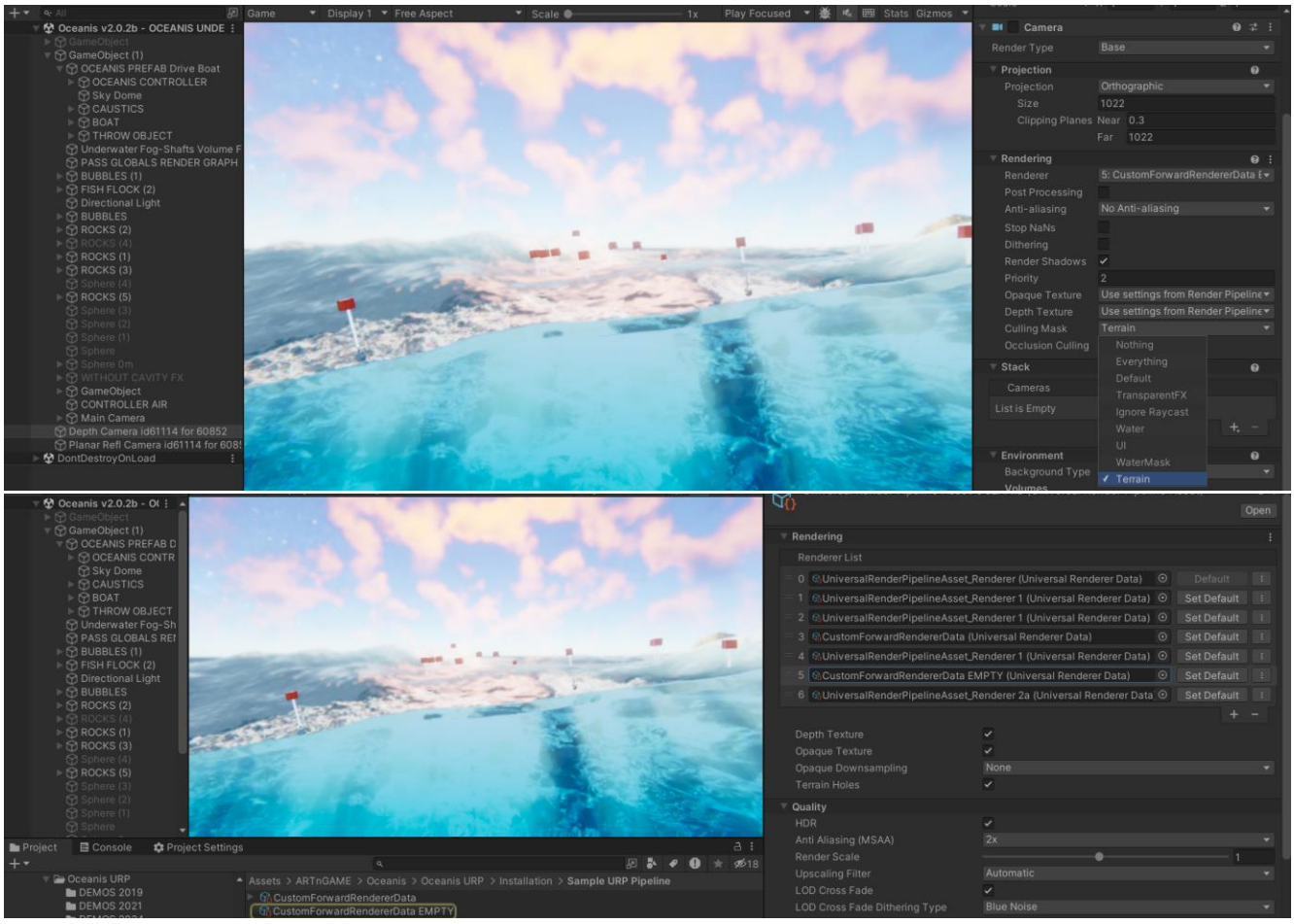
The main layers and tags to setup are the **"WaterMask"** layer in slot 8 that is used to render the water mask and the **"Terrain"** layer in slot 9, which is used to limit to certain objects the top down depth rendering that is used in the shore line system in the water module. Also can use the **"Terrain"** layer for the reflections control or add another layer that will have the reflection objects. The "**MainCamera**" tag should be added to the main camera of the scene.

The **“WaterMask”** layer is not hardcoded and can have any name or placement in the layers list, as long as the watermask camera renders the layer where the water mask mesh is on. This layer must be excluded from the main camera. The following images **show the layer applied to the water mask camera**, the **same layer applied to the water mask mesh** and **excluded from the main camera**. Any other layer can be used instead as needed.

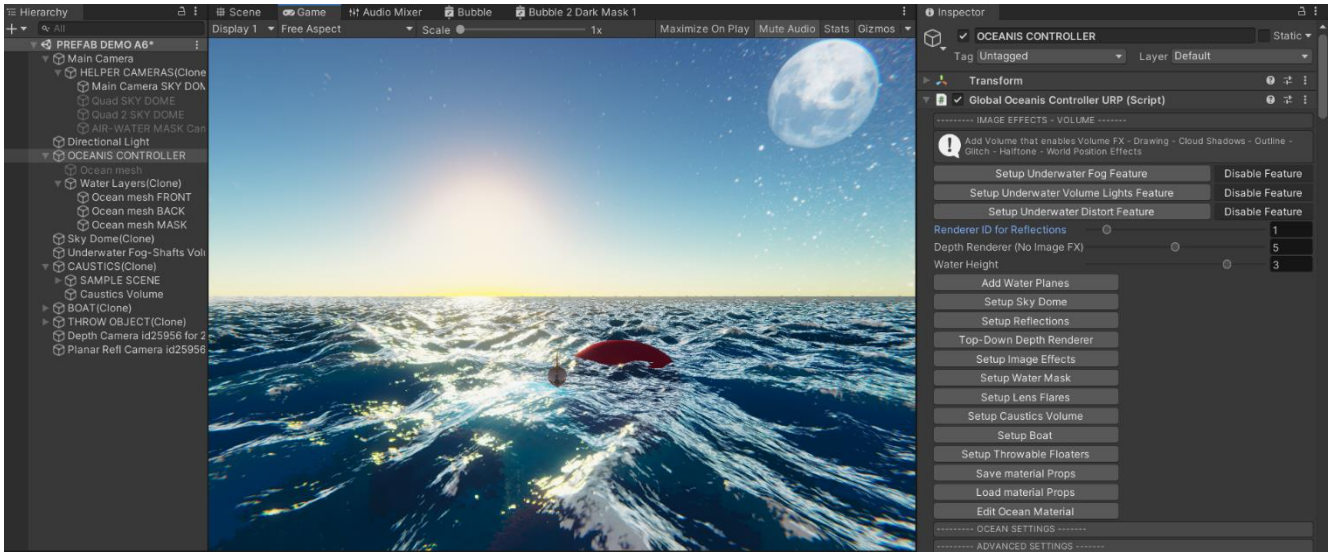


The terrain layer used for the top down depth renderer that handles the shorelines can also be any other layer than the “**Terrain**” one, and can be defined to the ocean system as shown below in the Depth Renderer section in the Oceanis controller.

This layer defined in the Depth Renderer settings section, will be automatically applied to the Depth Camera created by the system as shown below. The renderer of the Depth camera is defined in the above section in the “**Pass ID**” variable and should correspond to a forward render in the pipeline that has no image effects added, so the depth can be renderer correctly.



The main setup of the system is done by placing the "**GlobalOceanisControllerURP**" script in an empty gameobject in the scene. Then can add the modules step by step, by pressing the relevant buttons.



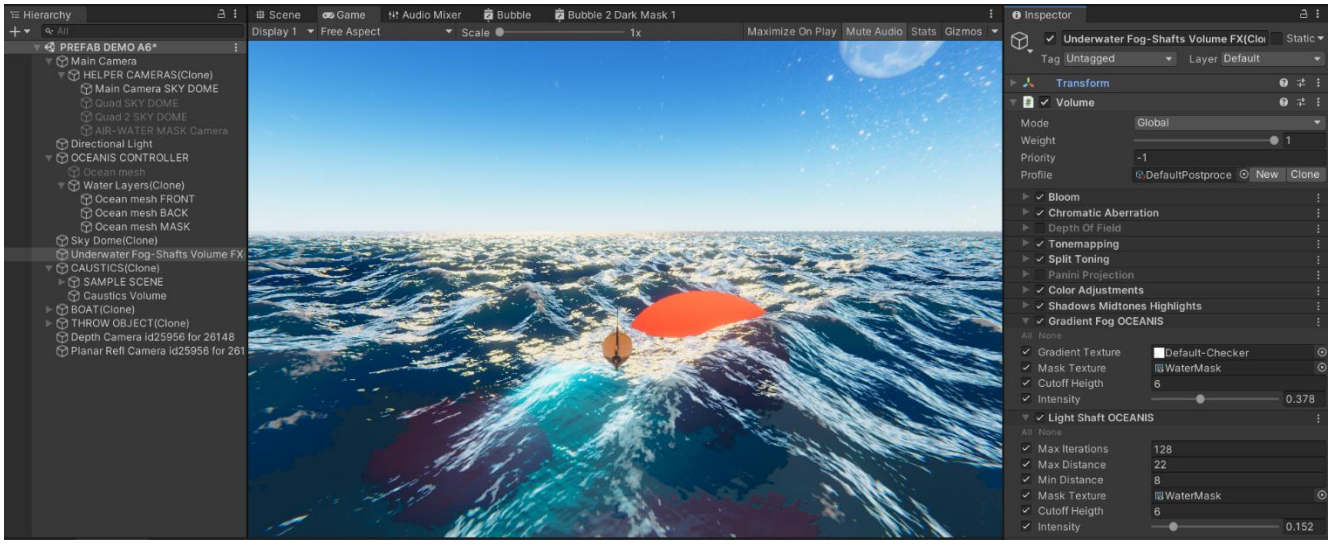
Setup Underwater Fog Feature

This button will assign to the Default renderer in the first slot in the URP pipeline renderers the needed renderer feature for the **underwater volumetric fog** module.

Setup Underwater Volume Lights Feature

This button will assign to the Default renderer in the first slot in the URP pipeline renderers the needed renderer feature for the **underwater Volume Lights** module.

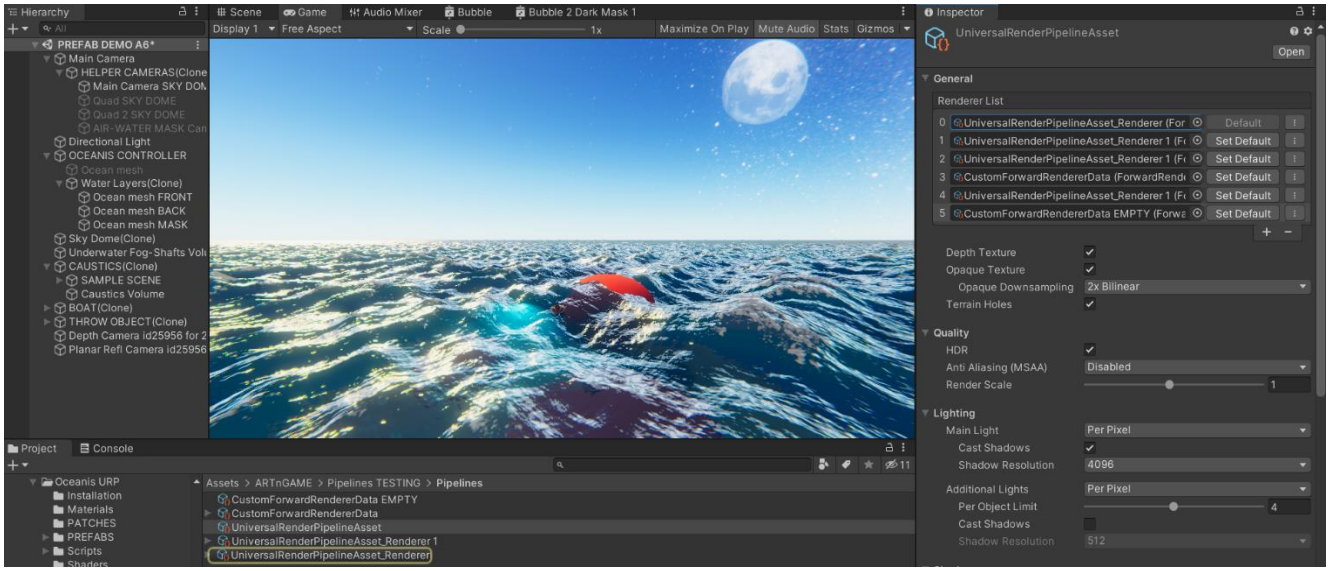
The above two image effects are also requiring a volume in scene to function, with the following entries as shown in the photo below. The effects should reference the water mask render texture.



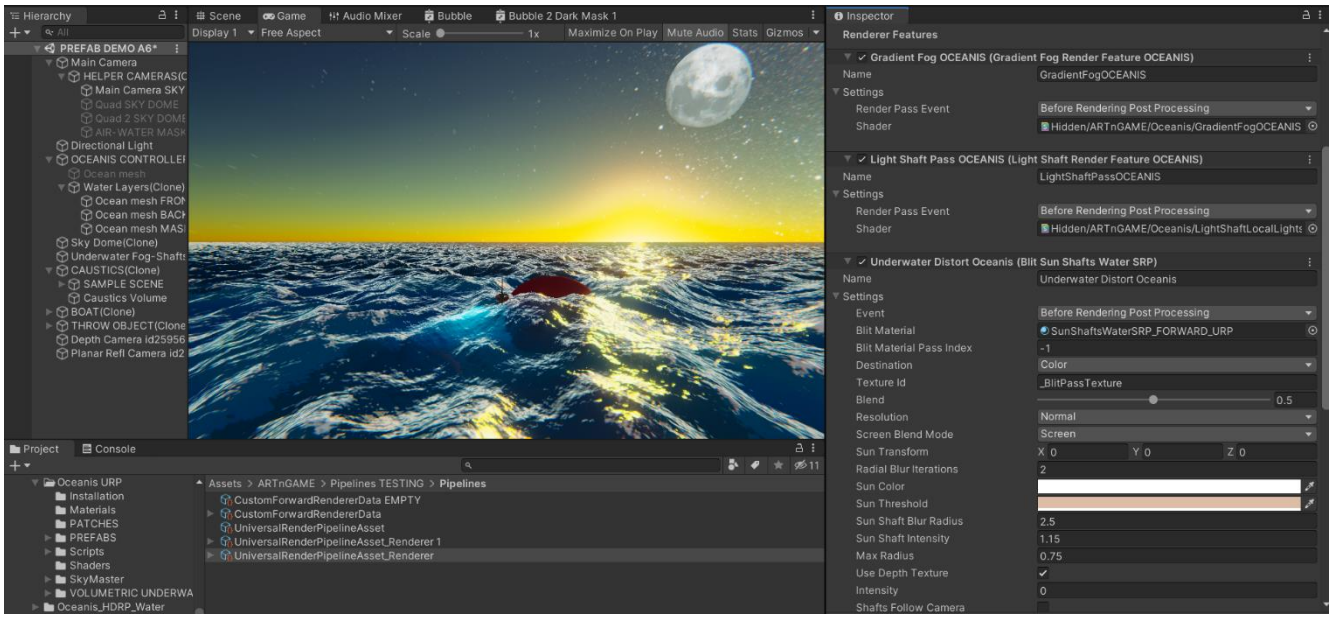
Setup Underwater Distort Feature

This button will assign to the Default renderer in the first slot in the URP pipeline renderers the needed renderer feature for the **Underwater Distortion** module, which allows rippling distortion in the underwater section, control of the line between water and air in various ways and other effects like Screen Space sun shafts.

An example of the pipeline used is shown below, note that both Depth Texture and Opaque Texture checkboxes must be activated.



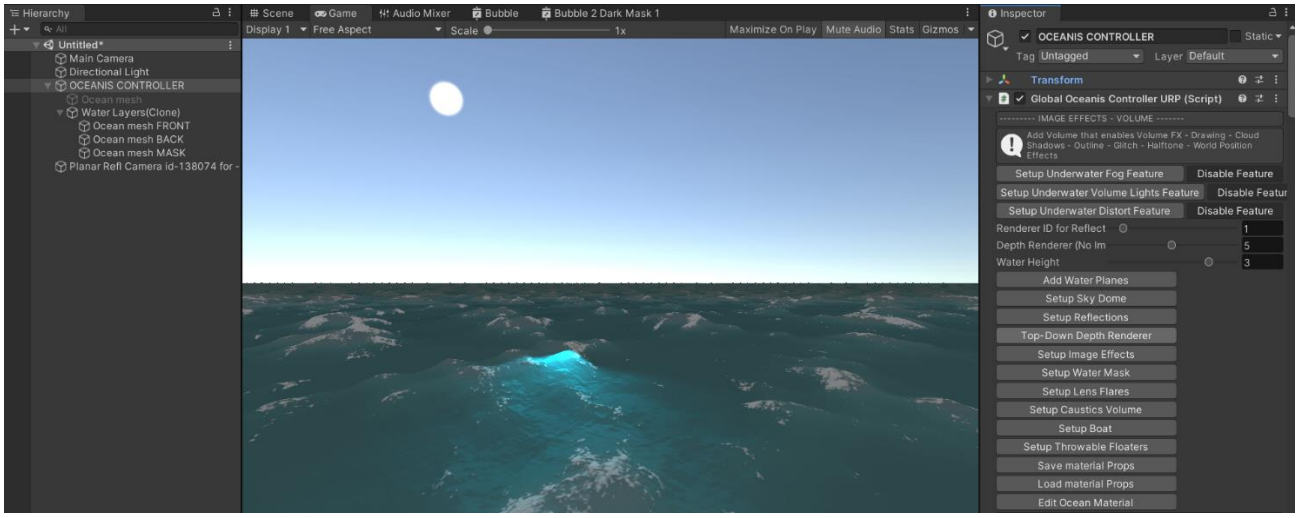
The forward renderer with the added renderer features for the underwater image FX is shown below.



The main setup of the features is in **the Shader for the first two effects** and in the **Blit Material slot for the last effect**, those are automatically handled when inserting the features through the global Oceanis configurator described above.

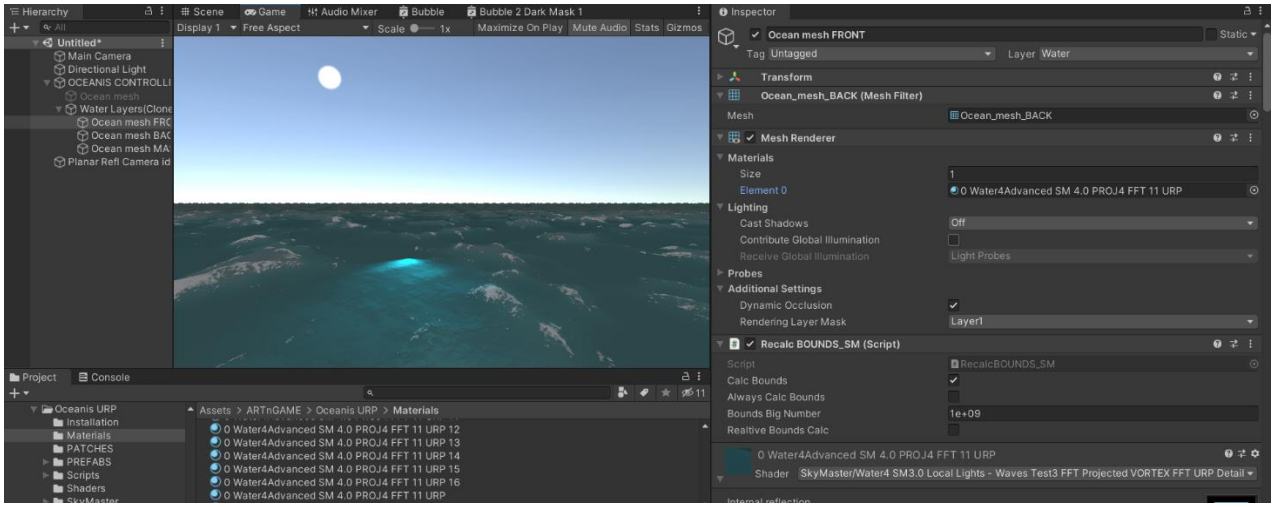
After adding the URP image effects in the pipeline, are now ready to do the setup of Oceanis in the scene using the **Global Oceanis Controller URP script.**

First define the reflections camera and top-down depth camera renderer IDs, that correspond to the URP pipeline renderers list. Ideally for Depth a renderer without any image FX assigned is required. For reflections can use any renderer, but in some cases should be a specific one that support for example cloud reflections, the system is directly compatible with **Sky Master ULTIMATE URP volumetric clouds, lighting and fog systems,** by applying the Sky Master image effects in the reflections renderer with the “isForReflections” checkbox activated. Example scenes for the users of both Oceanis URP and **Sky Master ULTIMATE URP** are available on request, from my internal project that combines the two systems**.**

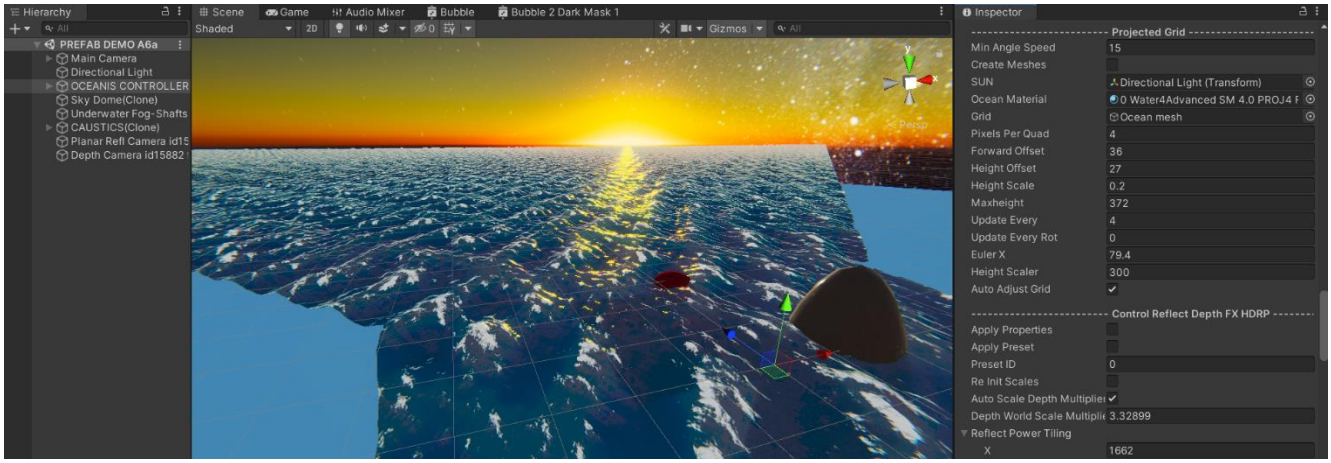


After define the renderers and desired water height in “Water Height” slider, press the “Add Water Planes” button to add the three water planes shown above, the FRONT, BACK and MASK layers. The mask layer is assigned to the “WaterMask” layer and the FRONT and BACK ones assigned to the “Water” layer.

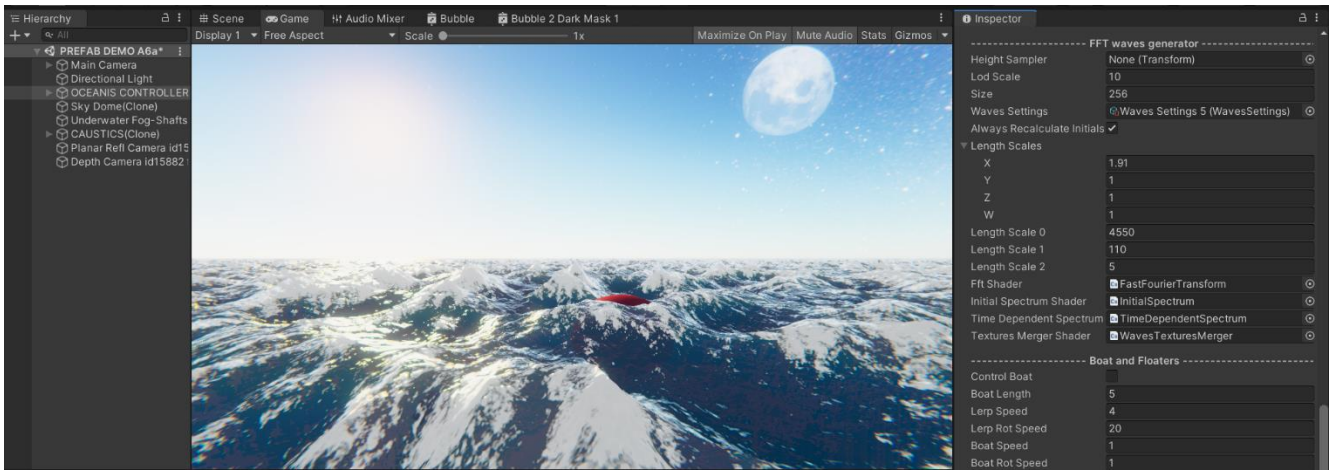
The planes have a “RecalcBOUNDS_SM” script attached that is used when the game starts to define the bounds to a large number so won’t be culled by Unity system. If the water layers not appear also try to press the “AlwaysCalcBounds” during editor time. The planes are assigned the main water material which is also referenced in the various scripts, the “0 Water4Advanced SM 4.0 PROJ4 FFT 11 URP” material. This material is used for rendering the water and its properties are set and saved per scene using the **Global Oceanis Controller URP script “Saved Props Material”.**



The planes are controlled by the projected grid system of the Oceanis, which is configured in the following section. The water mesh is created based on the camera view vector and position.

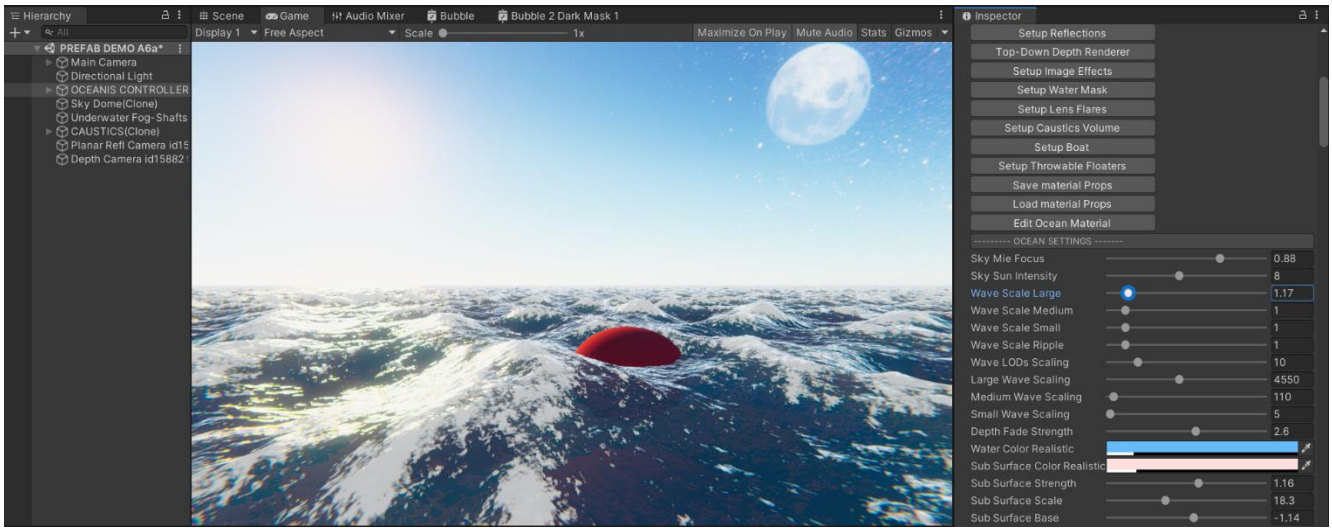


The planes material is taking as input a few textures with the Fast Fourier Transform (FFT) solution for emulating waves, the module that creates those textures is configured in the following section (“FFT Waves Generator”).

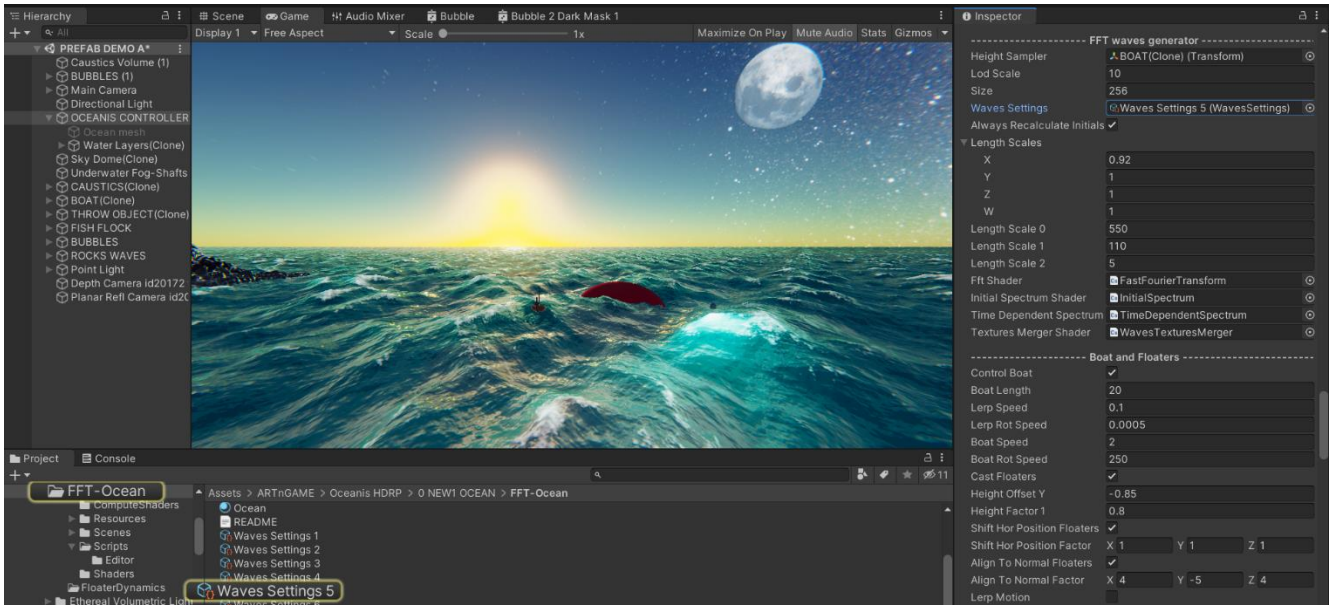


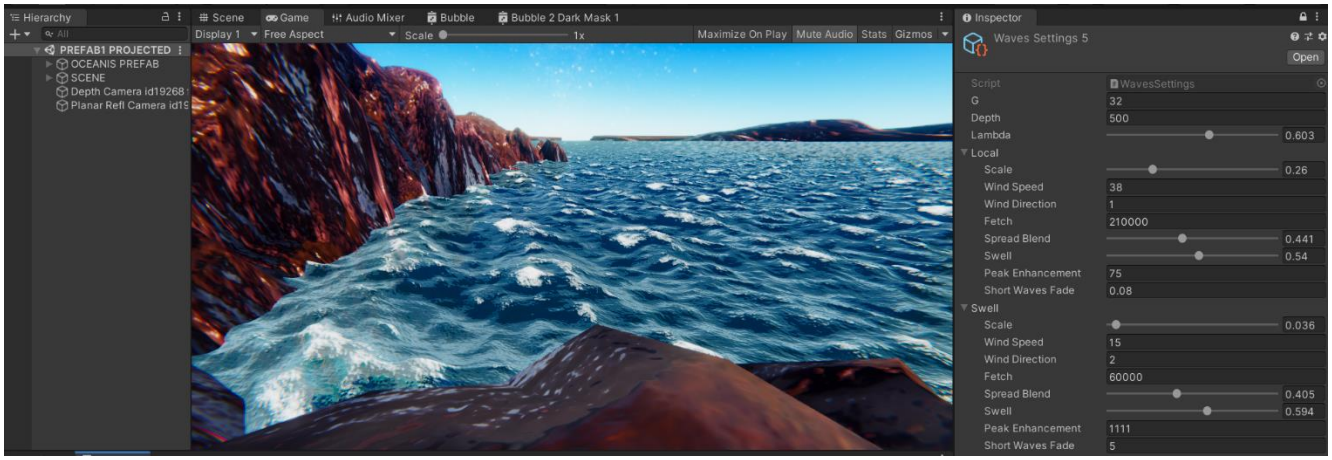
In this section can control the vertical scaling of the waves in “Length Scales” vector and the horizontal scaling in the “Length Scale 0-1-2” variables.

These options are also available directly in the Ocean material configurator section, shown in the image below, for easier access.



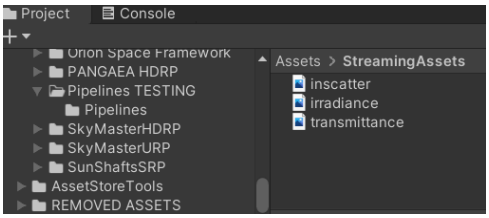
The waves can also be further refined by the options in the Wave Settings, as shown below



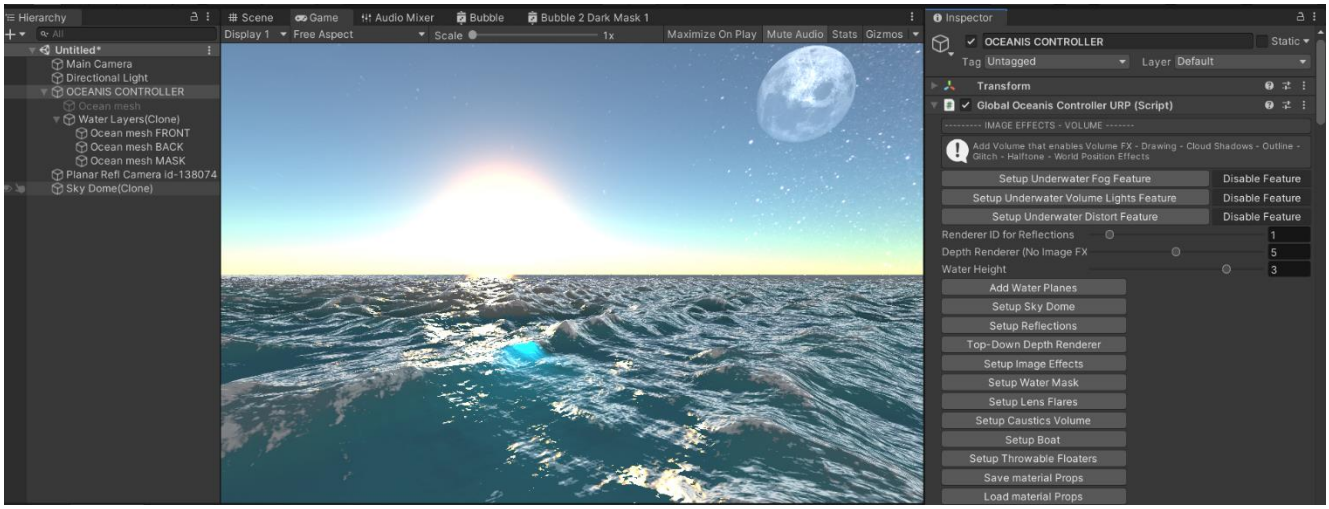


After the planes have been added, the next step is to configure the Sky module that creates a realistic sky rendering that is also assigned to the ocean for realistic fuzzy reflections.

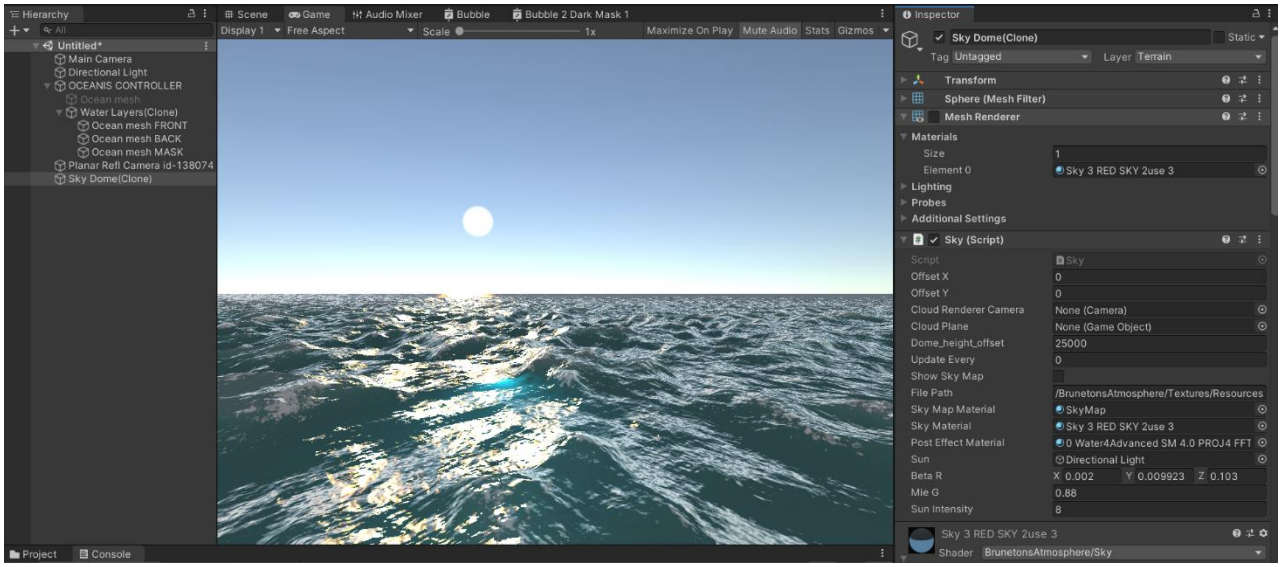
Note that this module uses the three files shown in the following image, that must be in the Streaming Assets root folder.



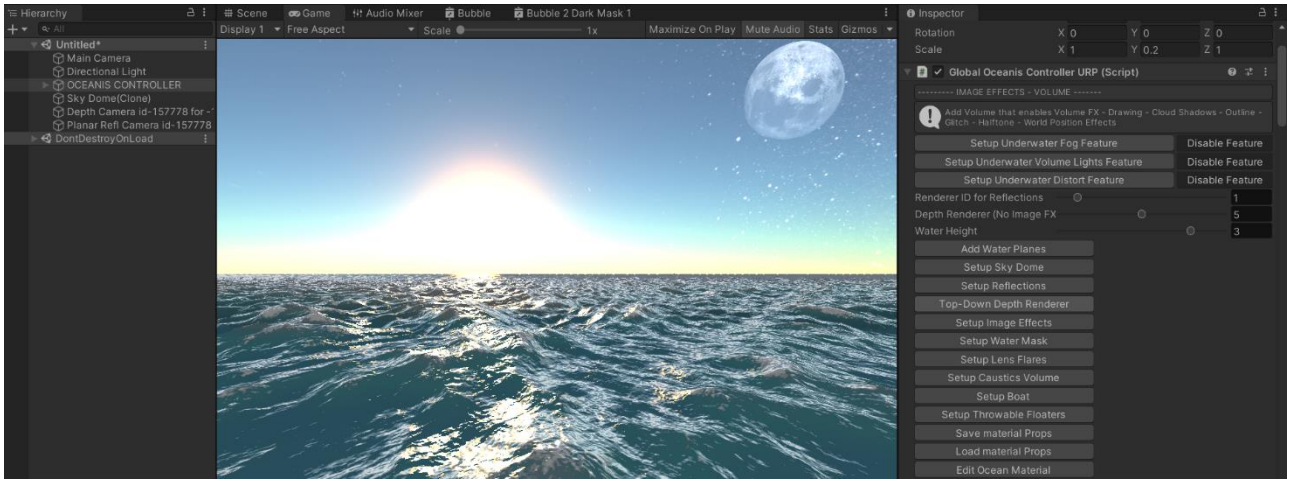
In order to add the Sky module, press the "**Setup Sky Dome**" button, which adds the sample sky dome and Sky control script.



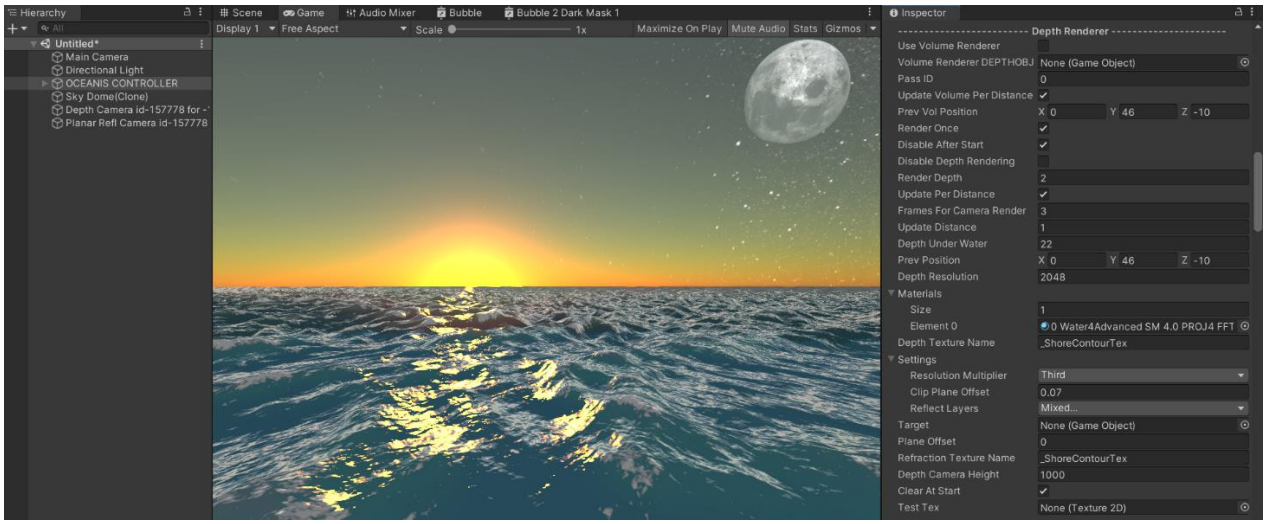
Note how the water takes now into account the Atmospheric Scattering rendering of the sky into account in the reflections for a realistic water rendering. The mesh of the sky dome may be turned off if the sky itself is not needed.



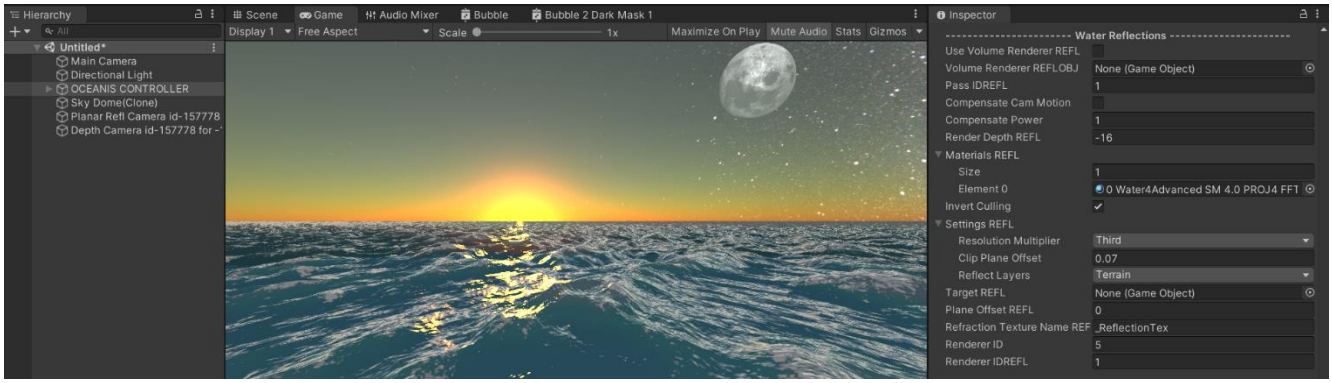
The above image shows the system using the Atmospheric Sky Scattering rendering on the ocean, with the actual sky dome disabled. The “Sky” script is responsible for the Atmospheric Sky Scattering calculations and can be tweaked for various sun scatter and color looks using the “**Beta R**”, “**Mie G**” and “**Sun Intensity**” variables. The “**Mie G**” concentrates the scattering around the sun making it more round as it gets higher towards one, the lower the value than one spreads the scattering more.



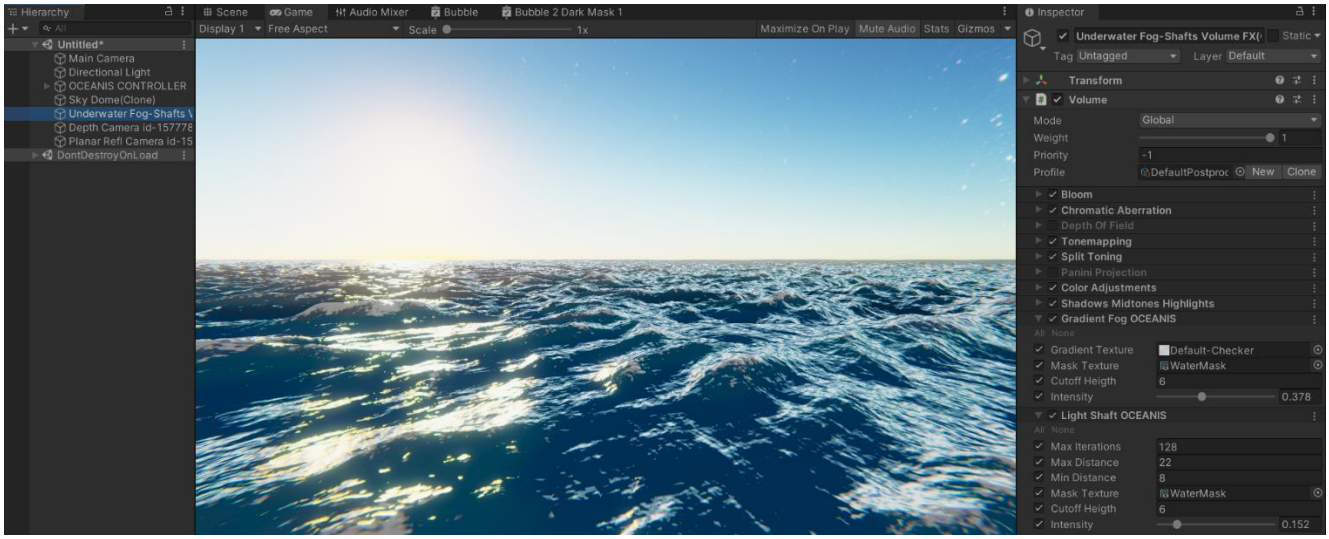
After the Sky has been added, the next step is to setup the Reflection and Top-Down depth rendering cameras, by pressing on the “Setup Reflections” and “Top-Down Depth Renderer” buttons. The later will enable the Depth Camera as seen on the hierarchy on the left panel in the above image. The Depth camera rendering options are changeable through the Depth Renderer section below.



The reflections rendering can be further tweaked in the “**Water Reflections**” section as shown below.

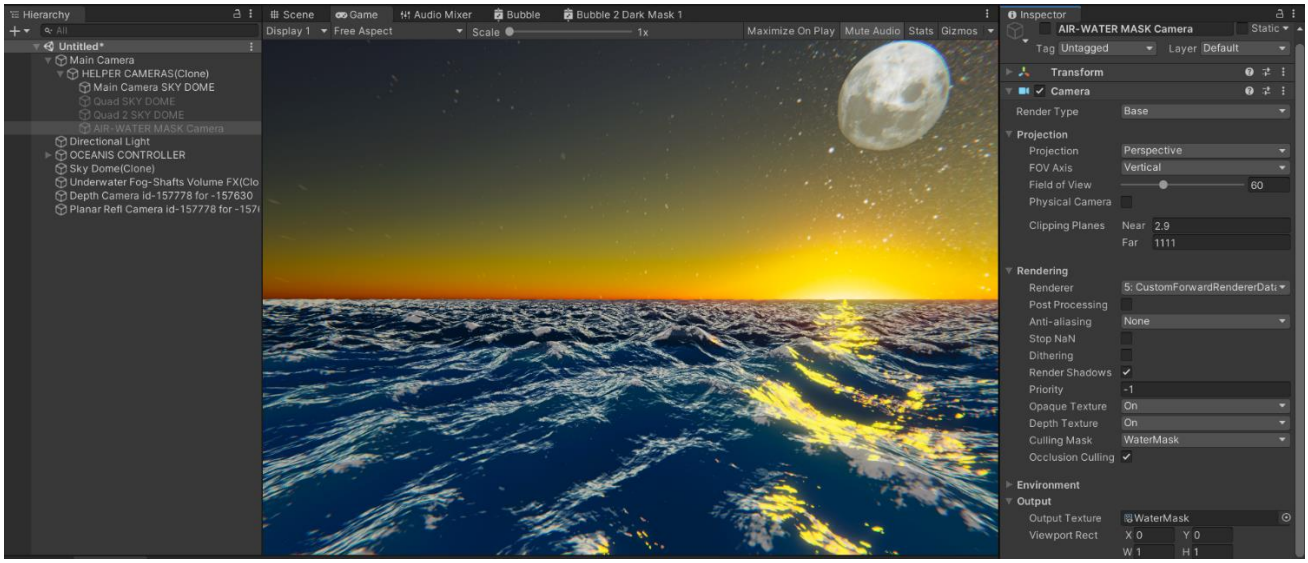


After the Sky – Reflections and Depth rendering, **the Image effects can be applied** by pressing the “**Setup Image Effects**” button. This will add the preset image effects, that are in the “**Underwater Fog-Shafts Volume FX**” object.

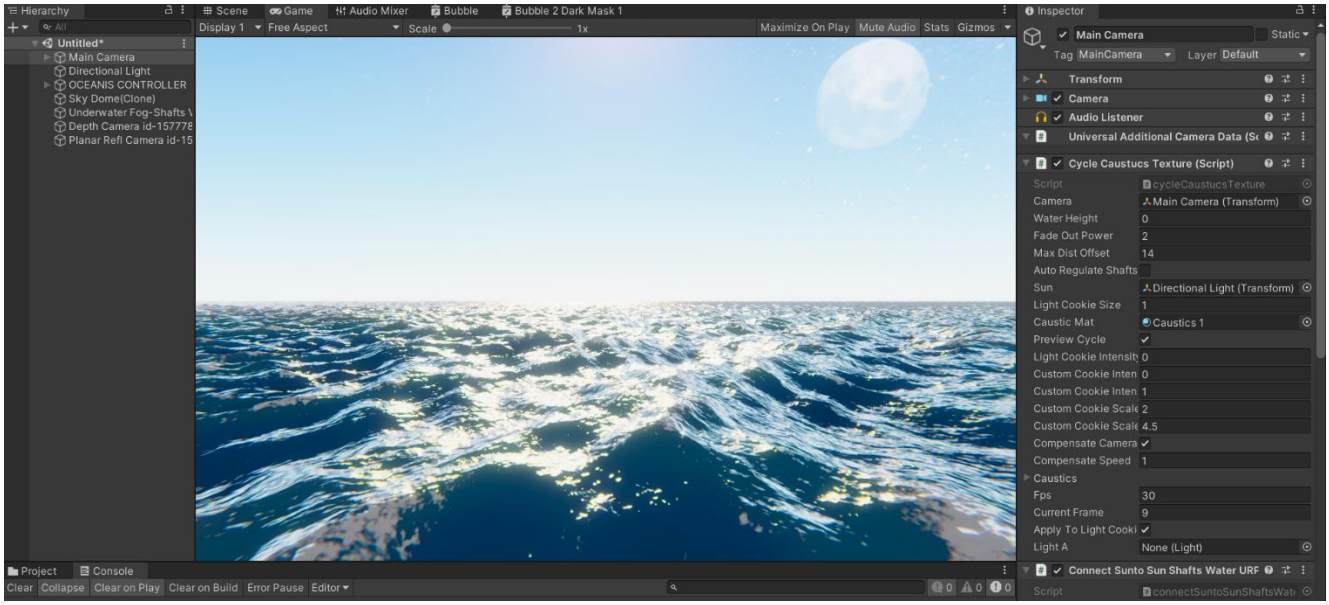


It is advised to first click on the Main Camera to activate the URP properties, so the image effects can be fully activated in the camera when pressing the "**Setup Image Effects**" button afterwards.

After the Image Effects Volume has been added**, the actual Underwater Image Effect can be activated** by pressing the "**Add Water Mask**" button. This will add the Water Mask rendering camera under the Main Camera so can follow the Main Camera movement, as shown below.



For the control of the image effects, two extra scripts are added to the camera. The "CycleCausticsTexture" controls the caustics texture cycling on the caustics module and also passes the caustic cycling textures to the underwater Volumetric Lighting system, so can create variable shafts based on the sun position. Note that the Volumetric Lighting system can also use the new in Unity 2021 light cookies system, controlled by the "LightCookieIntensity" variable, which is zero by default as cookies is not supported in all Unity and URP versions.

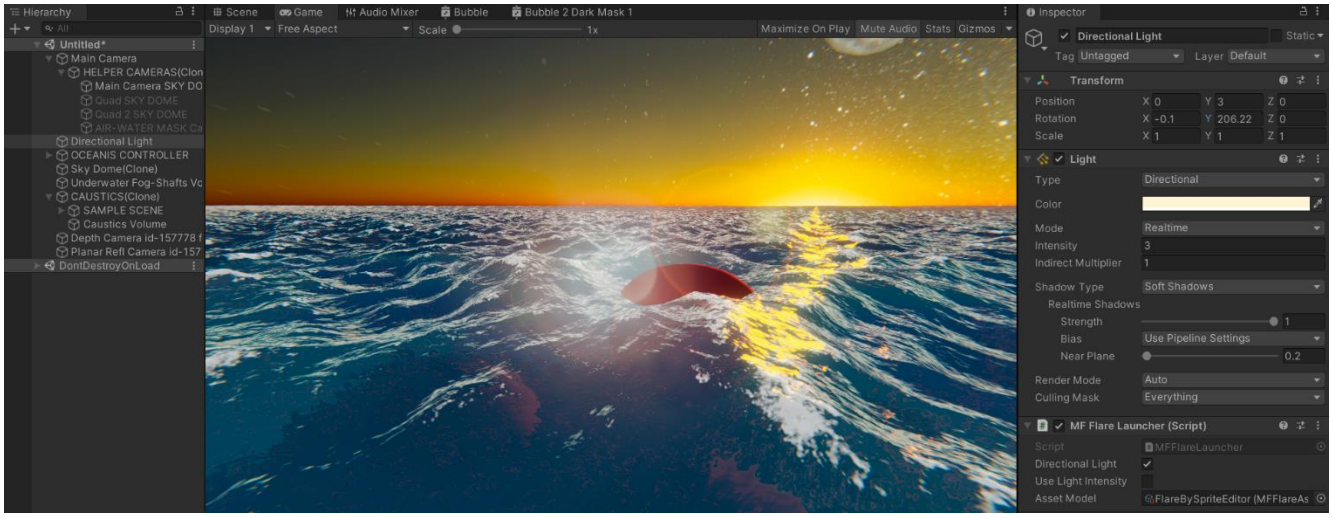


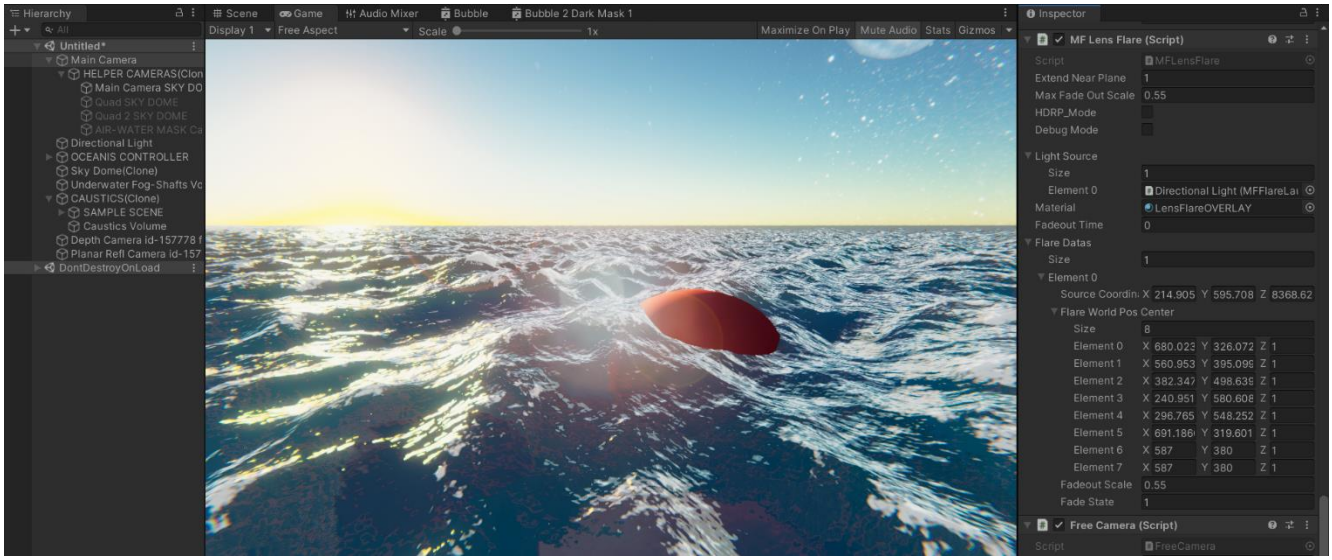
The other script added to the camera “ConnectSunToSunShaftsURP” is responsible for controlling the simpler Screen Space sun shafts effect and the underwater ripples and water-air line properties, width and other related variables.

Note that all the three modules (Distortion – Ripples, Volume Fog and Volume Lighting) must reference the “WaterMask” texture, for the Distortion – Ripples is referenced in the “**MaskMap**” slot shown above and for the Volume Fog and Volume Lighting in the relevant entries in the Image Effects Volume that was added in the add image effects step above.

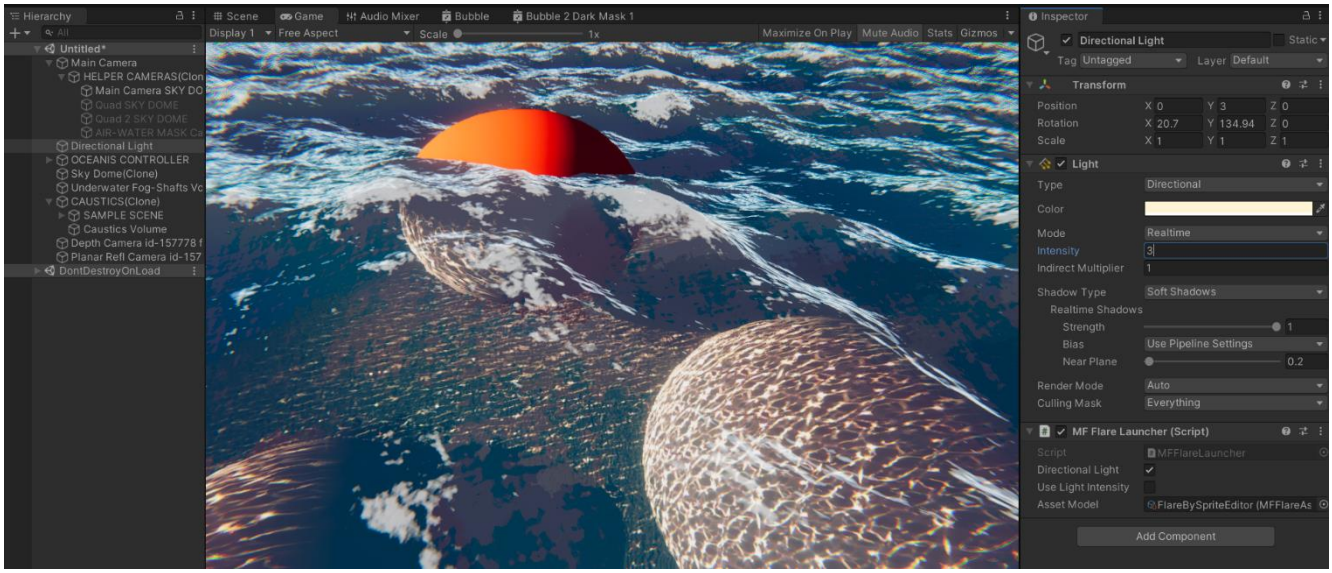
The above steps conclude the main water setup and extra modules like the caustics volume and lens flares can be added, also interactive objects samples with a boat and dynamic floaters that can rad the water heights and adapt their motion accordingly.

Press the “**Add Lens Flares**” button to add the lens flares system, this requires a script on the camera and on the sun light as shown below.

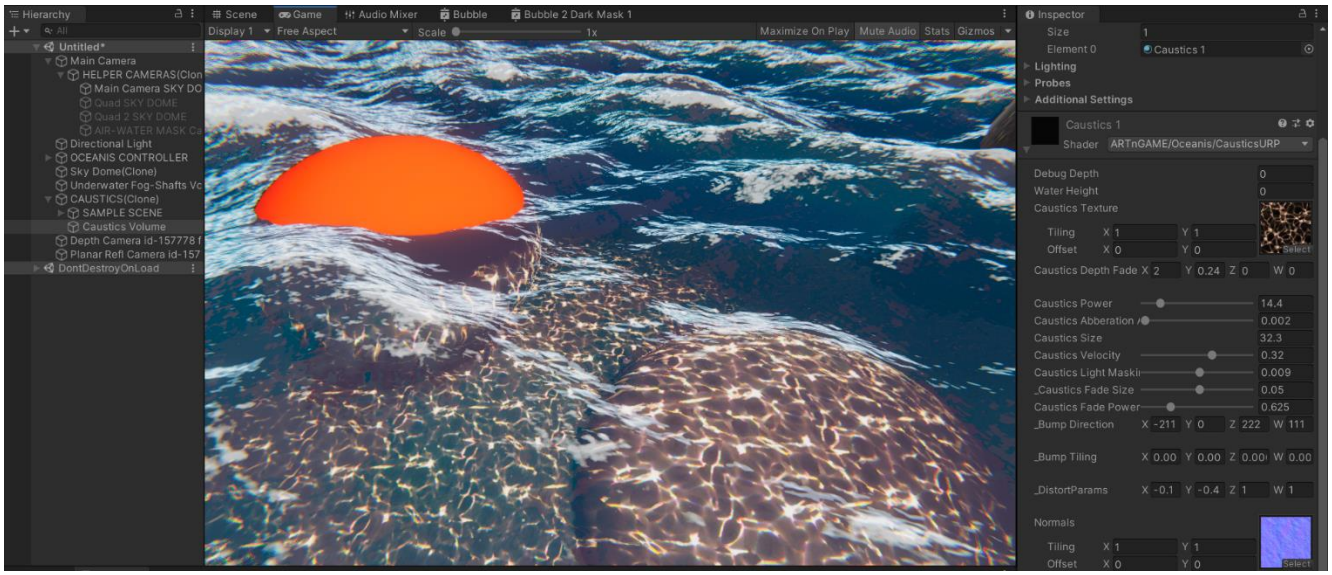

Hierarchy
Scene
Game
Audio Mixer
Bubble
Bubble 2 Dark Mask 1
Display 1
Free Aspect
Scale
1x
Maximize On Play
Mute Audio
Stats
Gizmos
Inspector
Directional Light
Static
Tag Untagged
Layer Default
Transform
Position X 0 Y 3 Z 0
Rotation X -0.1 Y 206.22 Z 0
Scale X 1 Y 1 Z 1
Light
Type Directional
Color
Mode Realtime
Intensity 3
Indirect Multiplier 1
Shadow Type Soft Shadows
Realtime Shadows
Strength 1
Bias Use Pipeline Settings
Near Plane 0.2
Render Mode Auto
Culling Mask Everything
MF Flare Launcher (Script)
Script MFFlareLauncher
Directional Light
Use Light Intensity
Asset Model FlareBySpriteEditor (MFFlareAs
Untitled*
Main Camera
HELPER CAMERAS(Clon
Main Camera SKY DO
Quad SKY DOME
Quad 2 SKY DOME
AIR-WATER MASK Ca
Directional Light
OCEANIS CONTROLLER
Sky Dome(Clone)
Underwater Fog-Shafts Vc
CAUSTICS(Clone)
SAMPLE SCENE
Caustics Volume
Depth Camera id-157778 f
Planar Refl Camera id-157
DontDestroyOnLoad


Hierarchy
Scene
Game
Audio Mixer
Bubble
Bubble 2 Dark Mask 1
Display 1
Free Aspect
Scale
1x
Maximize On Play
Mute Audio
Stats
Gizmos
Inspector
MF Lens Flare (Script)
Script MFLensFlare
Extend Near Plane 1
Max Fade Out Scale 0.55
HDRP_Mode
Debug Mode
Light Source
Size 1
Element 0 Directional Light (MFFlareLai
Material LensFlareOVERLAY
Fadeout Time 0
Flare Datas
Size 1
Element 0
Source Coordin X 214.905 Y 595.708 Z 8368.62
Flare World Pos Center
Size 8
Element 0 X 680.023 Y 326.072 Z 1
Element 1 X 560.953 Y 395.096 Z 1
Element 2 X 382.347 Y 498.639 Z 1
Element 3 X 240.951 Y 580.608 Z 1
Element 4 X 296.765 Y 548.252 Z 1
Element 5 X 691.186 Y 319.601 Z 1
Element 6 X 587 Y 380 Z 1
Element 7 X 587 Y 380 Z 1
Fadeout Scale 0.55
Fade State 1
Free Camera (Script)
Script FreeCamera
Untitled*
Main Camera
HELPER CAMERAS(Clon
Main Camera SKY DO
Quad SKY DOME
Quad 2 SKY DOME
AIR-WATER MASK Ca
Directional Light
OCEANIS CONTROLLER
Sky Dome(Clone)
Underwater Fog-Shafts Vc
CAUSTICS(Clone)
SAMPLE SCENE
Caustics Volume
Depth Camera id-157778 f
Planar Refl Camera id-157
DontDestroyOnLoad

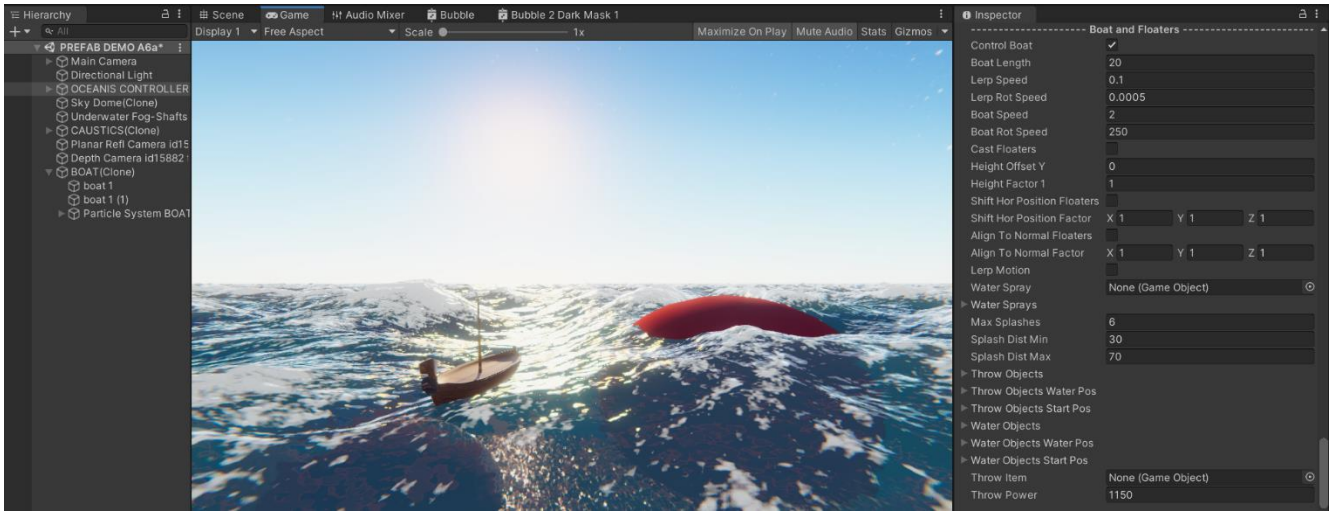
The “**Setup Caustics Volume**” button will add a caustics shader that is affected by the sun light power and direction.

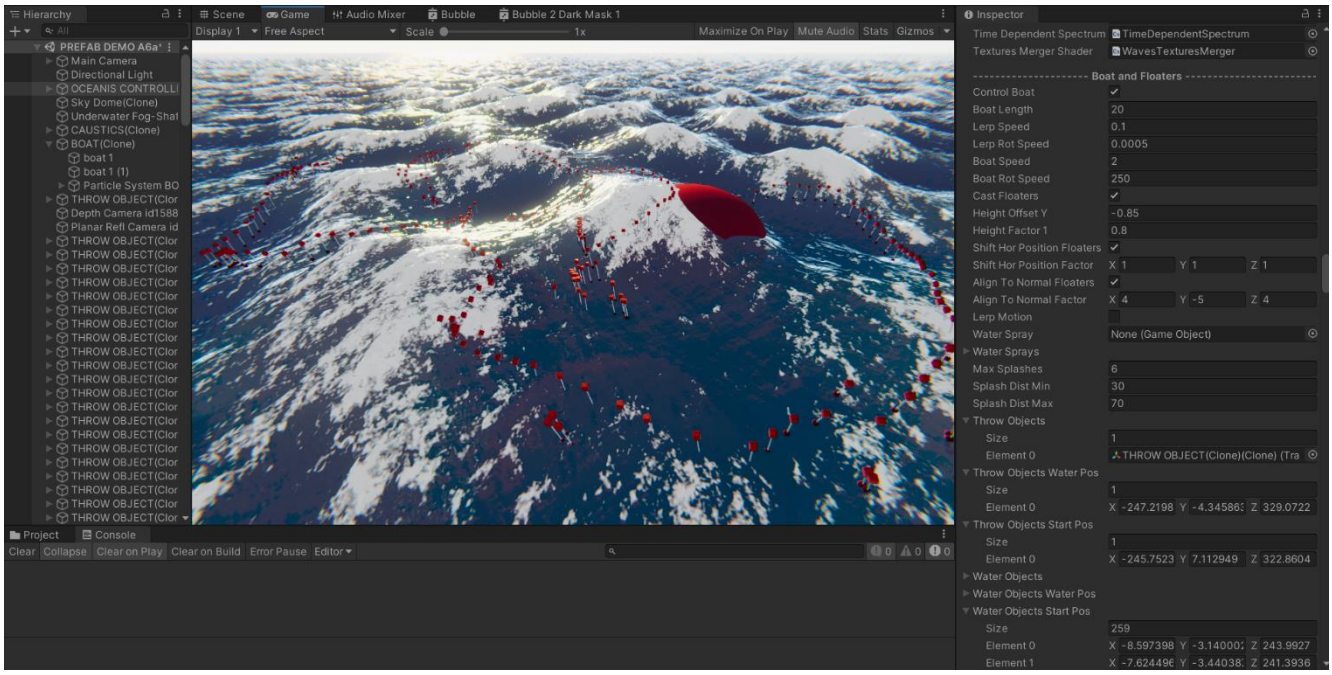


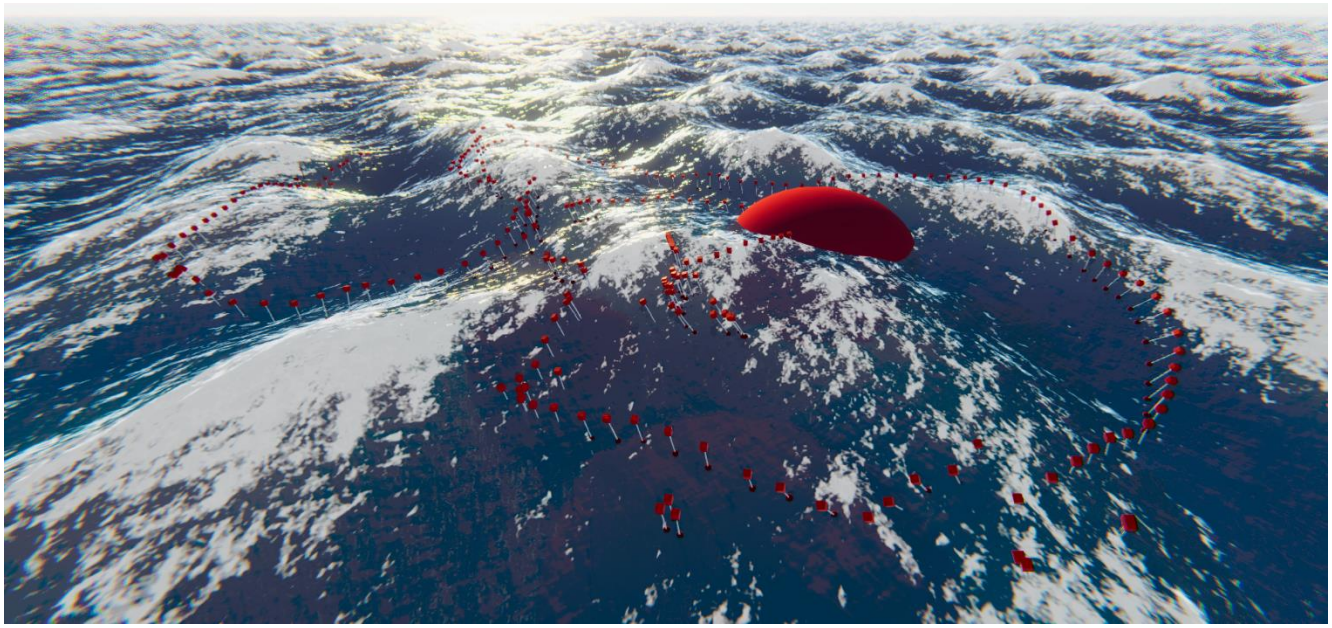
The caustics effect can be regulated directly in the caustics shader material, as shown below.



Boat and Floaters

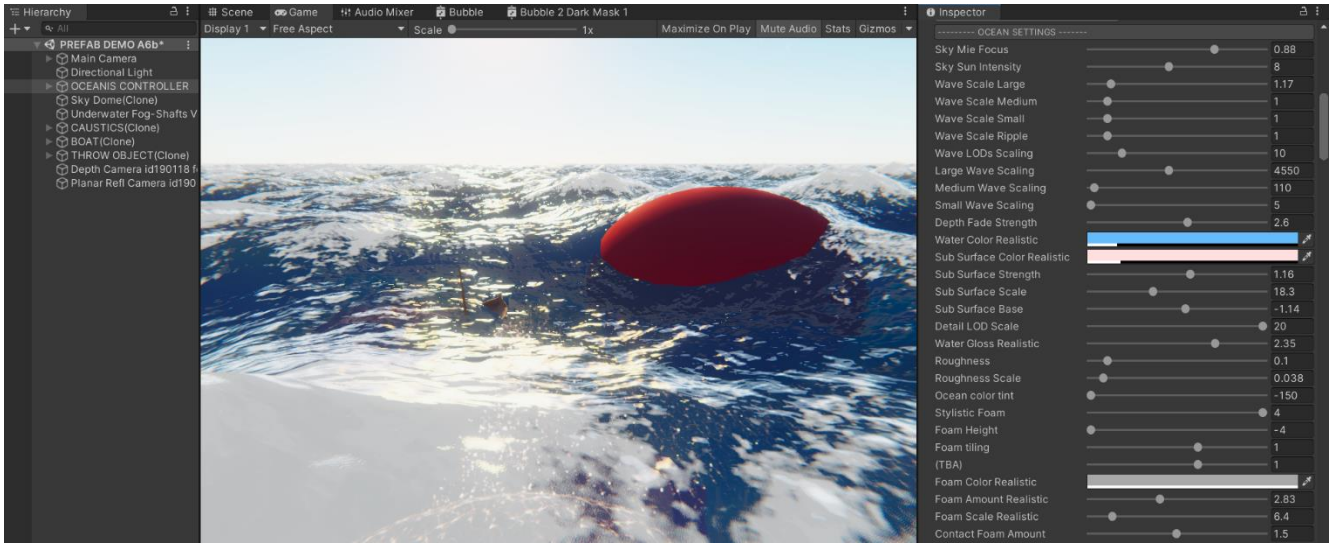


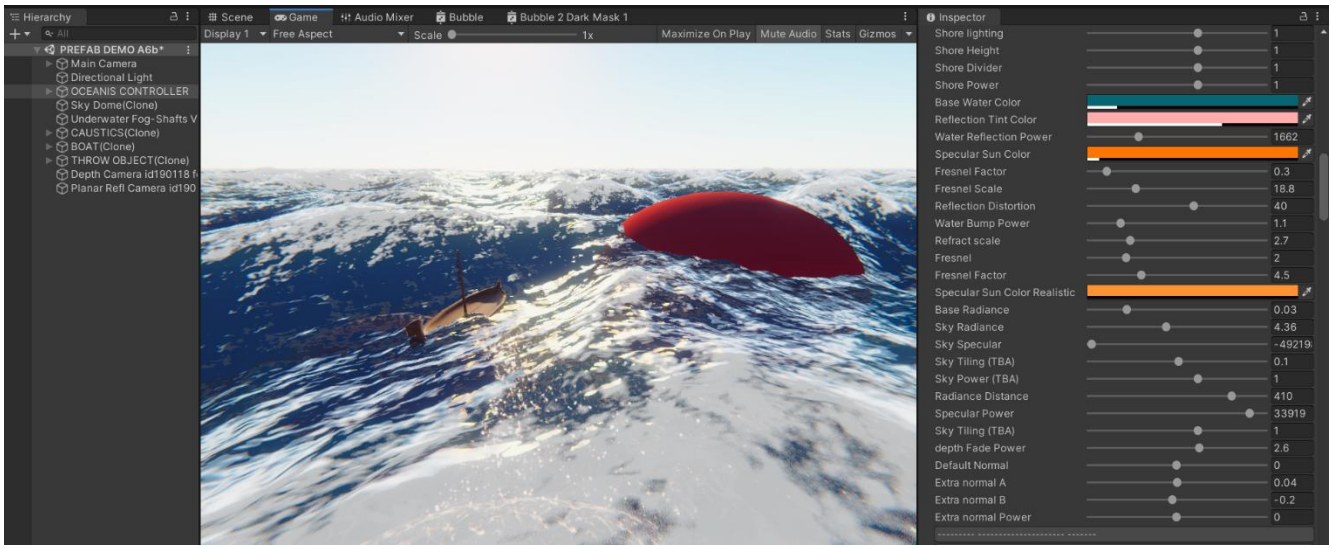




Ocean Material Editor

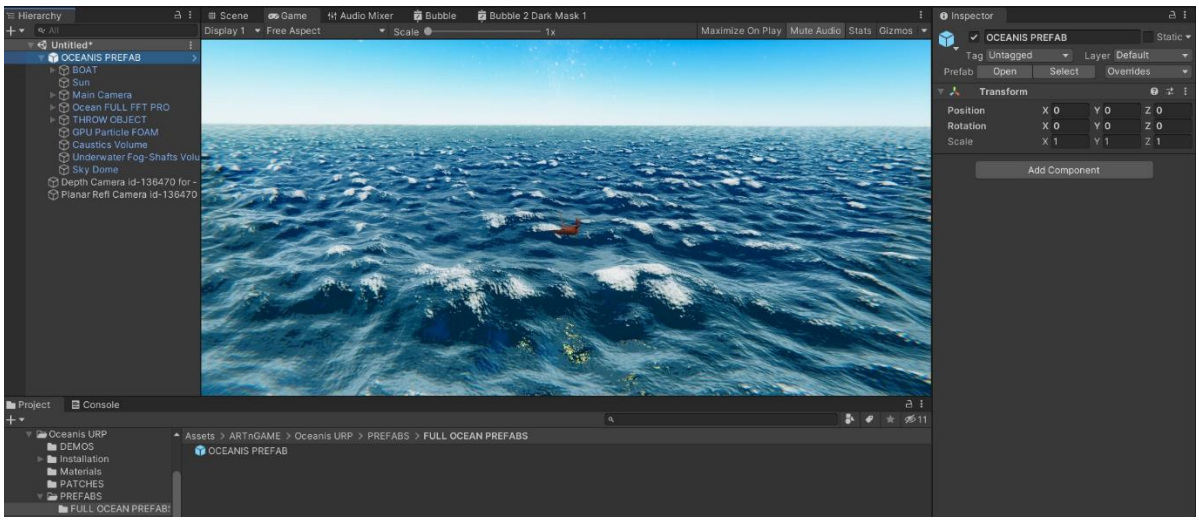
The ocean material properties can be changed in the global configurator as seen in the following





Setup with Prefabs

Alternatively to the step by step setup, can also use the ready prefab, the prefab contains all required scripts for the full function of the system. The step by step setup is recommended though if it is required to setup in a scene where a camera already exists and to choose the various modules.

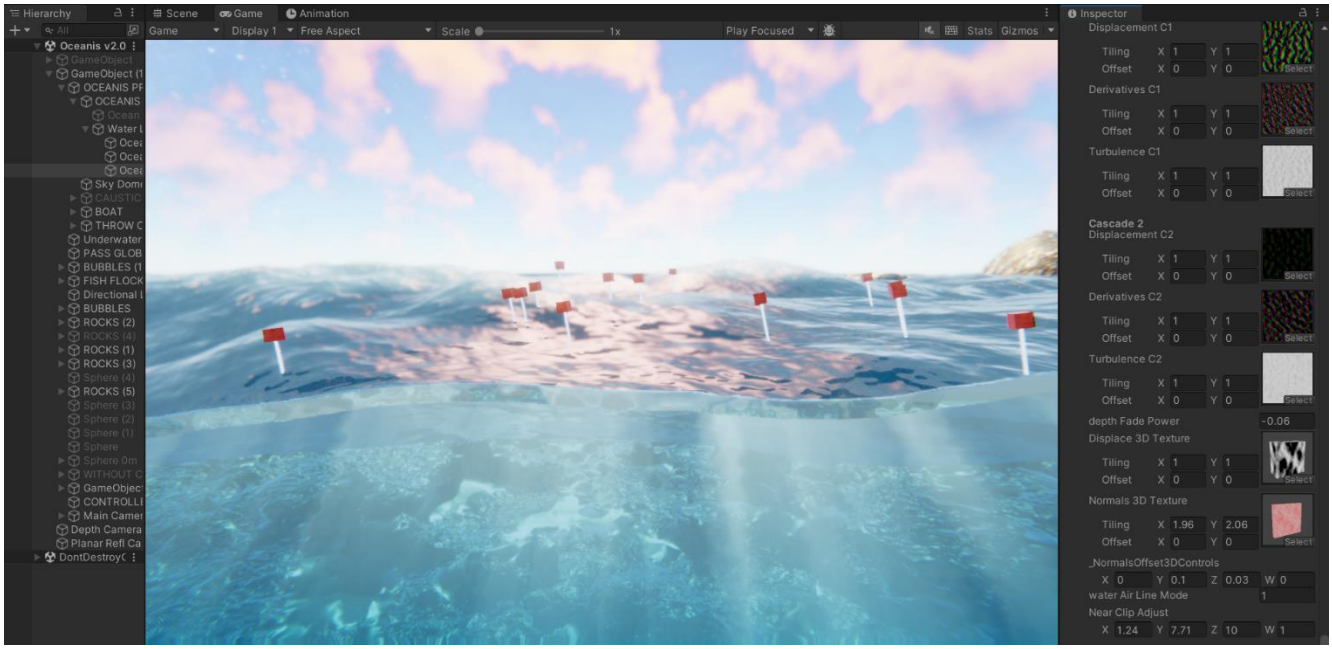


Oceanis 2024 New Modules and Modes

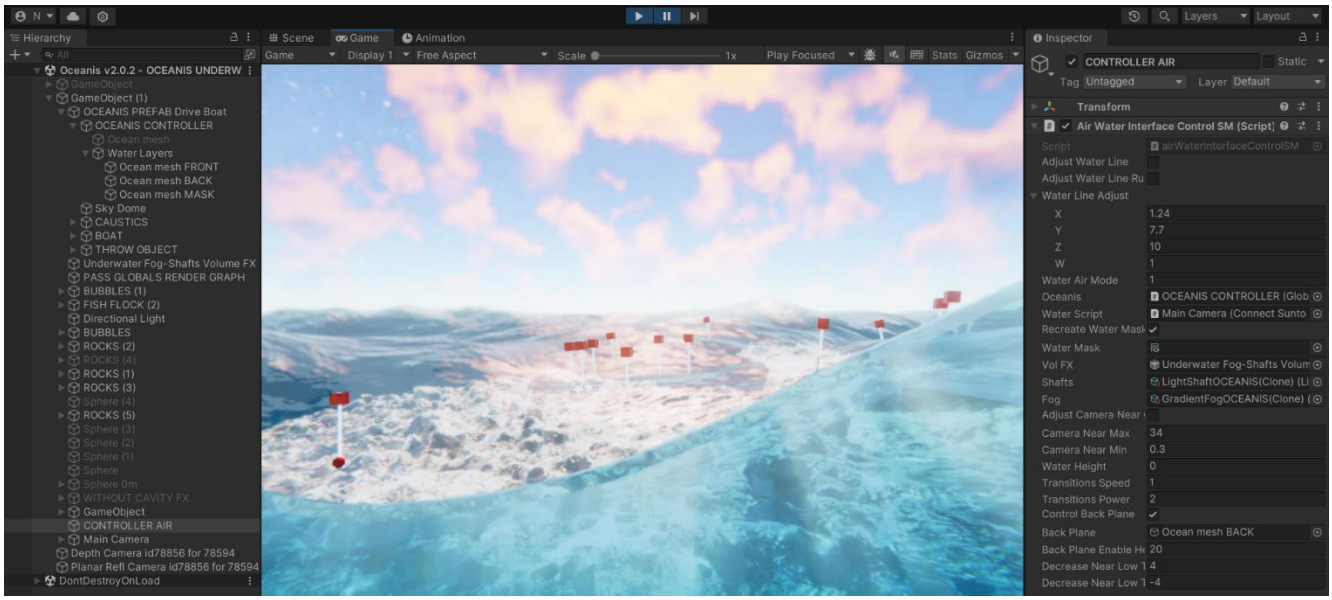
Oceanis 2024 contains a new mode for the air water line that is more precise and a new controller for tweaking the water line. Also a new dynamic ripples module and controller.

New air – water line mode and controller

The new water line mode is enabled in the following ocean material parameter "**Water air line mode**", when set to one will activate the new mode and zero is the old mode. In the new mode can control the effect by the "**Near Clip Adjust**" vector variable.



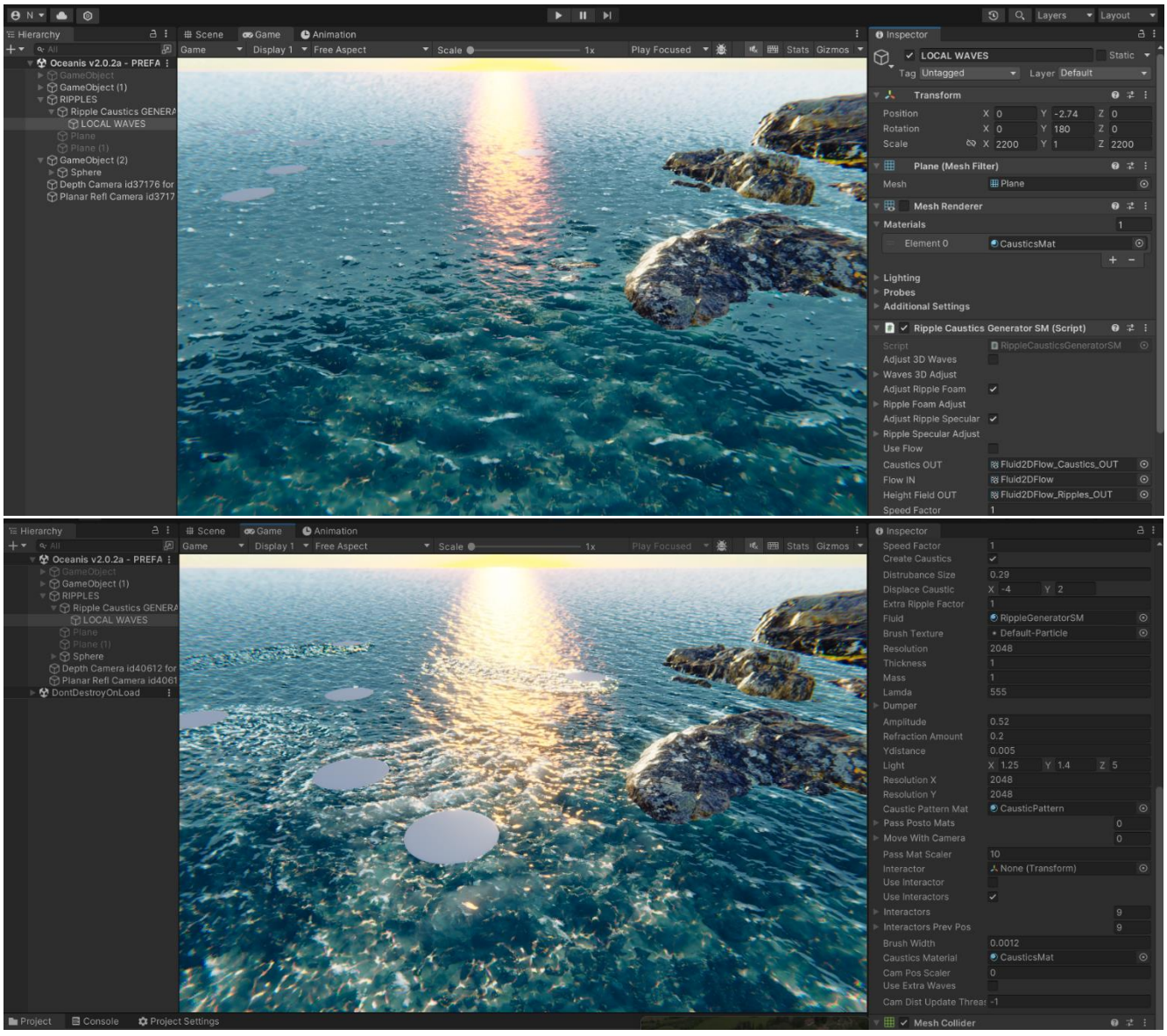
The system contains a scene controller that can adjust those variables and save the settings per scene. Otherwise if no controller is used, can use a unique copy of the ocean material and after any changes on the material itself should use the Save Material option in the Ocean controller to save the changes in the scene, though it is advised to use the new controller in each scene instead.



The controller must reference the Ocean controller, water line image effects controller on camera, in Water script and the VolumeFX that holds the underwater volumetric fog and sun shafts. Also can control the ocean back plane for fine tune its enable region. To enable the air line controls can use the two checkbox on top to enable in editor and play mode. Also can control the camera near FOV to give a clear cut in the water when near the surface and decrease the close culling when above or below water, so objects are not cut off early in the regions that is not needed.

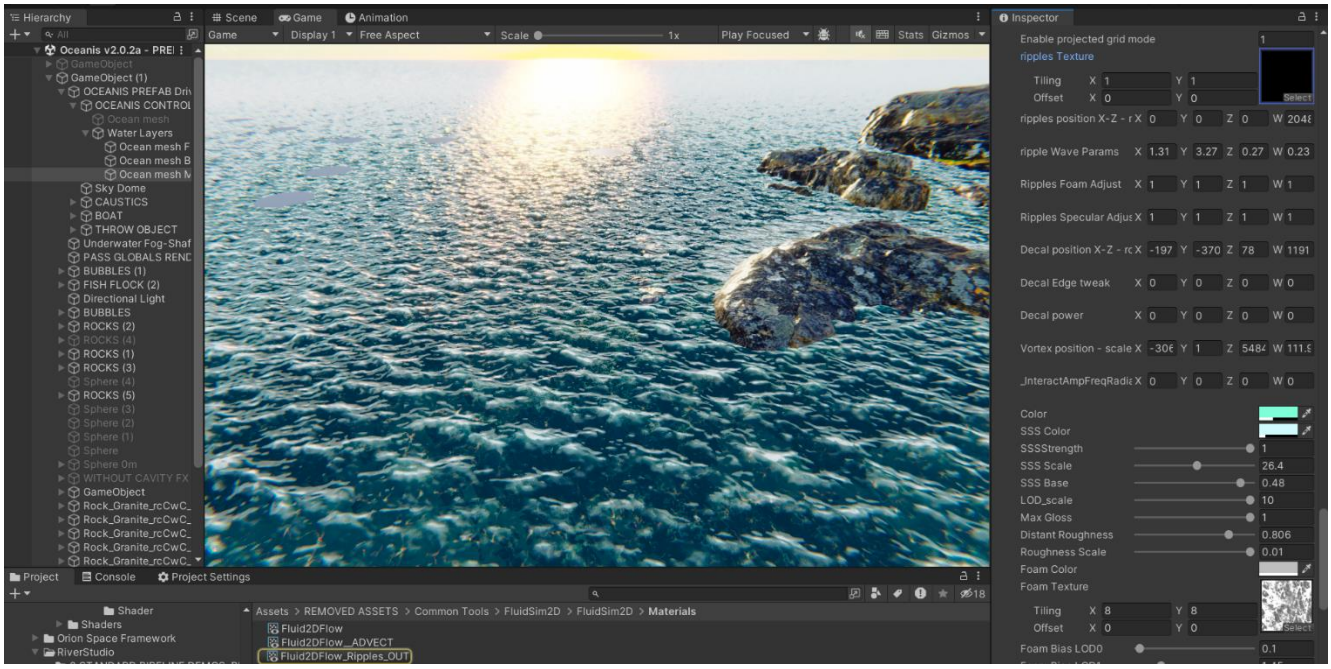
New dynamic water ripples - caustics and controller

The new ripples system allows for dynamic interaction with the water from the mouse and defined agents (Video). The system is comprised of a ripple creation script shown in the below image and a collider plane that senses the collisions of mouse and agents.



The interactors can be declared in the “**Interactors**” list and will create ripples on their motion. The brush width defined the width of the interaction wave and Caustics material can be defined so can receive the dynamic caustics from the system that correspond to the dynamic ripples. The “**Lamda**” defines the staying power of the ripples and “**Speed Factor**” the fade out of the ripples and how many steps are calculated per frame. The “**Resolution X-Y**” define the caustics texture resolution and “**Resolution**” defines the ripples render texture resolution. The “**Light**”, “**YDistance**” and “**Refraction amount**” control the caustics creation and lighting.

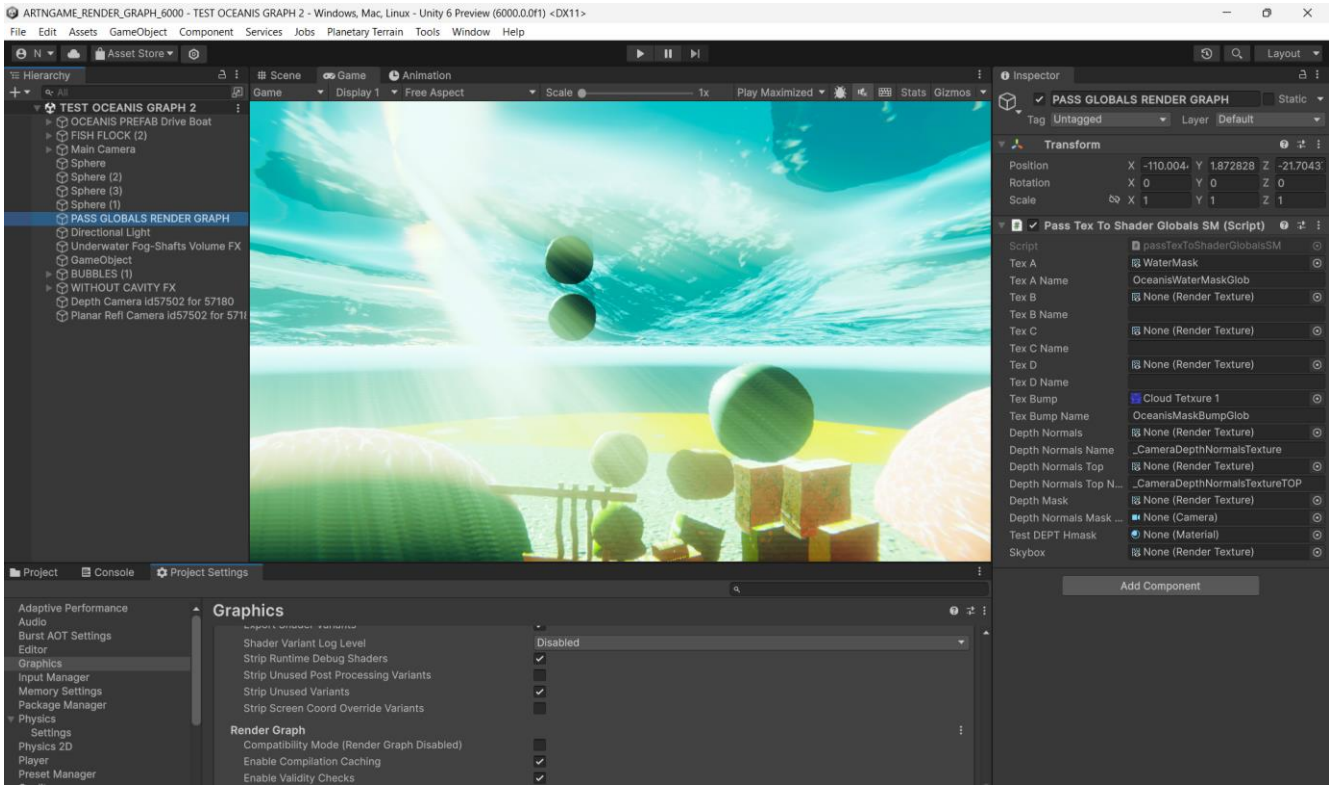
The Ocean material must be referenced in the “**PassPostoMats**” in order to automatically setup the material parameters based on the scaling of the “**LOCAL WAVES**” item with the collider that defines the region of interaction. This way can scale the system and move it around the map and will adjust the Ocean material parameters accordingly in the below section, in “**Ripples position**” vector. Also can enable the control of the “**Ripple wave params**”, “**Foam Adjust**” and “**Specular adjust**” vectors, that fine tune the application features of the ripple texture solution on the ocean surface.



The system also contains multiple new demo scenes of the new systems in action, as shown below with ripples, caustics and wave interaction from multiple agents.

Unity 6 URP Render Graph support

Oceanis 2024 has support for the new **URP Render Graph platform**, using the below "**Pass Tex to Shader Globals SM**" script in order to pass some of the needed variables for the system to function properly, the **WaterMask** render texture and **bump map of water line** texture. Note that the URP Render Graph is in Beta, so can change radically in next Unity versions until the Unity 6 LTS release and is recommended to use the compatibility mode in Unity 6 instead for production.



The various image effects involved in the underwater are all compatible with the new Render Graph and the code supports both Unity 2022 and Unity 6 in a single module.