Rendering an Ocean with Displacement

This tutorial covers how to render an ocean scene using nothing but a polygon plane and an HDR image connected to a Skydome light. The detail for the ocean shader will come from a vector displacement map. There are a variety of attributes to set when using displacement mapping. However, it is very simple to setup and get convincing results using the right settings.

The ocean displacement map can be downloaded here.

The scene file is available here.

- Start off by creating a polygon plane that can be used as the ocean surface:
Ocean Material

- Create a **Standard Surface** shader and assign it to the polygon plane. Set the **Diffuse** to 0. Increase the **Specular Weight** to around 1. Increase the **Transmission Weight** to 1 (this will make the water transparent). Finally set the **index of refraction** to that of water **1.33**.

Vector Displacement Map

The vector displacement map used is a 32-bit EXR image. This means it is able to retain the highest quality for the vector displacement effect. To get a more realistic wave effect, we will use a vector displacement map to displace the sea waves. This will displace the sea geometry around a direction different from the normal, unlike a traditional displacement map which displaces the geometry in one direction only.

If the effect of the waves looks too sharp you could try blurring the vector displacement map in order to get smoother waves.
• Connect the vector map to a Maya displacement node and connect that to the shading group of the Ai Standard used as the sea material. The shader network should consist of the vector displacement map being connected to the Maya displacement node which is connected to the shading group node of the sea material.

Ai Skydome light

Create a Skydome light and connect an HDR map to its Color attribute.

Subdivision settings

• Render the scene. There is no displacement! That is because we don't have enough subdivisions in the geometry of the sea plane.

• Select the plane and under the Arnold attributes, change the Subdivision type to Catclark and increase the number of iterations. Start off with a low number like 3:
The sea looks ok, but it really needs more detail. Increase the **Subdivision Iterations** to **8**. Be careful not to increase this value too much as memory use and render times can increase dramatically with each increment (each iteration quadruples the geometry).

**Lighting**

- Finally, add a **directional light** to the scene. We can use this to exaggerate the effect of the sun on the sea surface, thereby bringing out the highlights on the sea surface.
• Increasing the **Angle** attribute of the directional light will increase the angular size of the light, thereby producing a larger specular reflection on the sea surface.

**Transmission**

Light that penetrates the ocean surface is 'refracted' due to the fact that light travels faster in the air than in water. When the light reaches the water it will be scattered or absorbed by particles in the ocean. As the ocean becomes deeper (denser), the water will decrease the depth of light penetration. We can replicate this effect quite easily using the **Transmission** attribute in the **Standard Surface** shader. The effect can be
seen when placing an object inside the ocean surface. We can see that the longer light travels inside the mesh, the more it is affected by the blue *Transmission Color*.

![Transmission Color: White vs. Blue](image)

When applied to the ocean shader we can see the effect that *Transmission Color* and *Depth* has on the white cylinder inside the closed surface ocean below:

![Ocean Shader Effect](image)

**Dark Patches**

The ocean shader is not fully transparent. This means that non-glancing rays will not reflect and will instead refract and go through the water (like in real life), but since it's not transparent, we instead terminate the
rays and get black.

Shading normals can cause problems when the shading normal points away from the ray even though the geometric normal points towards the ray. In this case, the shading (Auto Bump) normal says that we are hitting the backside of the water and should reflect down, while of course in actuality the ray is hitting the top of the water mesh.

At grazing angles real water is nearly 100% reflective, but even so, it may take several bounces for the light to get out of the valley behind a wave crest, and even after several bounces it may reflect something that is higher up from the horizon than what the surrounding water reflects.

A more realistic look is to make the water transparent and then place a plane below the water that has the color of ocean water. In the example images below the ocean floor’s Standard shader has an Emission Scale of 1 and a blue/green Emission Color. Increasing the Subdivision Iterations has also helped to reduce the number of black patches on the displaced sea waves.

![Subdivision Iterations 3. Black patches are visible](image1.png) ![Increasing the Subdivision Iterations to 8 reduces the problem](image2.png)

The dark patches behind the wave crests (self-reflection) can also be improved by increasing the Specular Ray Depth in the Render Settings.
Conclusion

That concludes this tutorial on how to create a believable ocean scene. Using the correct displacement settings in conjunction with a high quality 32-bit vector displacement and the correct Standard shader settings should give you something that looks realistic. Why not try adding depth of field or change the camera lens to something like a fish eye in order to add more visual interest to the scene.

Below you can see how much more detail can be achieved by using a vector displacement map to create the wave effects compared to using regular geometry:
Animation of displacement value

Further examples