uWater - Fast Shaders

Multiple lights even on mobile devices - How can that be a fast shader?

Instead of using expensive per-pixel lights, we decided to stick to vertex lighting. On the upside vertex lighting is a lot faster than per pixel lighting. On the downside the water mesh needs to be sub-divided.

Especially on mobile devices more triangles are less expensive than complex shaders – of course depending on the situation.

Be reasonable with the triangle count though! The more resolution your mesh has, the sharper your light spots can be.

We do not spend performance on specular light calculations. This shader is limited to reflections and diffuse illumination. At the moment we do not see any useful way to integrate per-vertex specular.

If you do not need point or spot lights in your scene, there is no need for sub division. Simple enable Per-Pixel Fresnel and you can use simple quads.

Supported light types:

Directional Light

-Works best and does not even need a sub-divided mesh

Point Light

-Needs a sub-divided mesh

-The less triangles per m^2 you have, the more diffuse the light will behave

-Needs a lot of triangles for really sharp lights

Spot Light

-Similar to the Point Light

-Light Cookies will not work(!)

-Stick to directional and point lights if possible

Using the Per-Pixel Fresnel

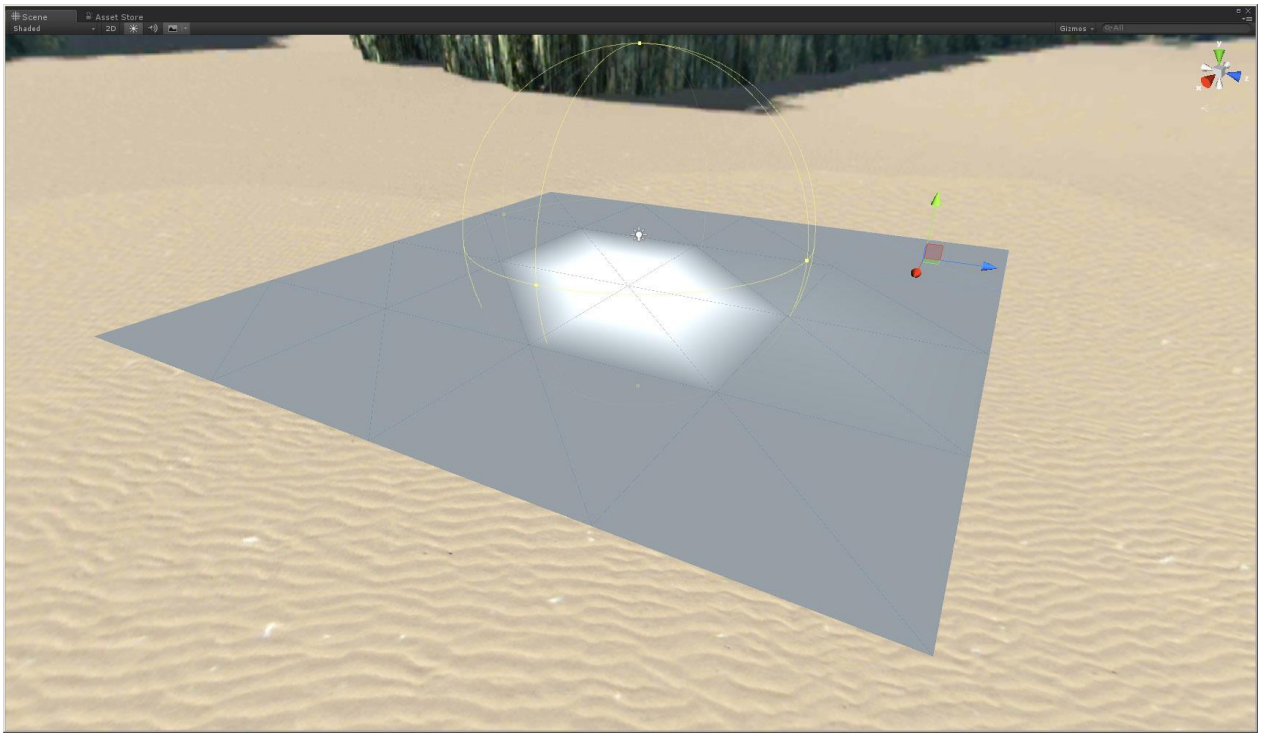
This is a powerful option. You need to decide if you either want to reduce the triangles of the water mesh, or make the fragment shader less expensive on the GPU.

If enabled the shader uses the more expensive per-pixel fresnel instead of the computationally cheaper per-vertex fresnel. The more verticies your water mesh has, the more accurate the fresnel effect will be. Keep in mind that more verticies also cause more vertex shader work.

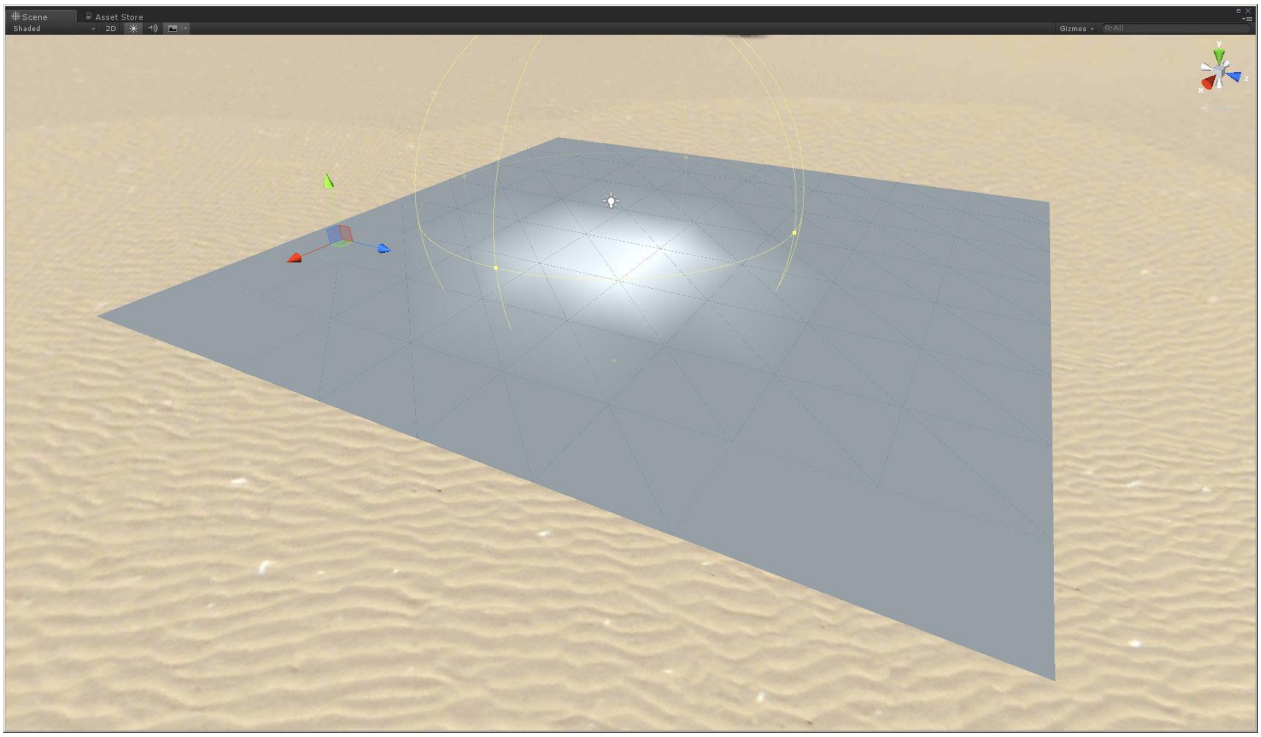
How to set up the geometry

Vertex lighting generally works very well if you stick to a few rules.

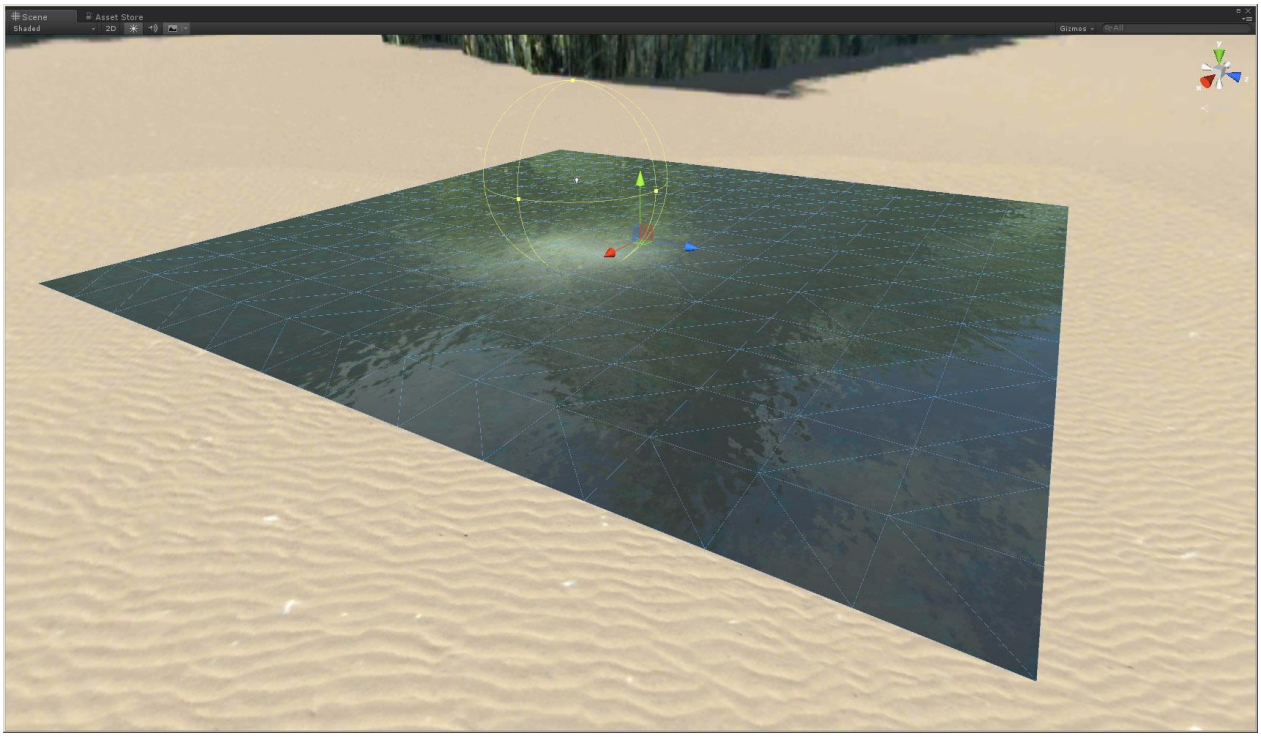
The worst case scenario is a high intensity light with a small range on a hardly sub-divided mesh. The light will not move smooth over the surface, but “jump” from one vertex point to another.



Slightly more divisions already smoothen the highlight a lot.



With the uWater shader, the issue is hardly visible anymore.



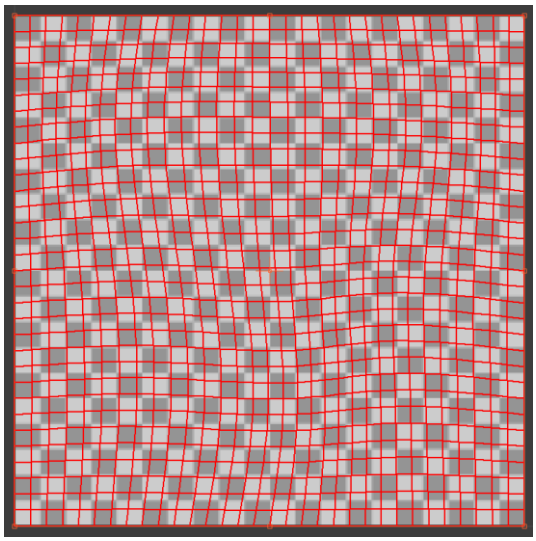
Better tiling and water flow

Tiling has always been a problem for water maps. Especially when seen from above the tiling becomes terribly obvious.

Short of making the shader a lot more complex there is nothing we can do to fix this. With the focus on mobile devives and maximum compability with older PCs that is not an option.

There is a neat trick to hide the tiling a bit, while making the flow of the water less linear.

This works best in combination with a sub-dived mesh. So if your mesh is already sub-divided to make better use of the vertex lighting and the per-vertex fresnel, you will get improved visuals without any performance hit.



Future Plans

-Transparent water

-Testing per vertex-specular

-A fully per-pixel lit version for deferred shading

-Foam and shore effects

-Water depth