In this tutorial we will take you through the process of rendering a head model from nothing but a sphere and a vector displacement map. Vector displacement maps are a powerful and efficient way to add detail to your scene without the overhead of managing complex geometry. They provide a very efficient workflow when working with models from 3D sculpting applications and Arnold. Unlike traditional grey scale displacement maps which simply use a line long for the vertex normal, vector displacement maps have RGB values which represent the XYZ displacement direction and size. This means they don't follow a fixed surface normal and can move in any direction, including overlapping each other.

However, vector displacement maps have larger file sizes because they have to store more information than regular displacement maps. Also bear in mind that vector displacement maps, as with normal displacement maps, are not perfect for every situation. For example, UV seams can be problematic when using vector displacement maps.

Many thanks to the very talented Zeno Pelgrims for providing the assets and assistance for this tutorial.

The sphere.obj, vector displacement map and diffuse color map can be downloaded here.
Mudbox

The original head sculpt has been extracted from Mudbox using the following settings.

- The vector displacement was extracted from the original head model in Mudbox under the menu UVs & Maps > Extract Texture Maps > New Operation
Mudbox provides the following settings for extracting textures from models. In this case we chose **Vector Displacement Map** and changed the **Vector Displacement** to **Absolute Tangent**. Ensure that the map is saved using **32-bit FP** depth and has a high enough resolution. This will give us an accurately displaced render in Arnold.
Make sure that you use a 32bit or 16bit floating-point format to store your image, and not an integer format. An integer format will not work correctly. This is because integer formats do not support negative pixel values, which are used by floating-point displacement maps.

The Utility shader can be useful for diagnosing issues when rendering displacement maps.