Pixel Style 2D Water Shader Guide
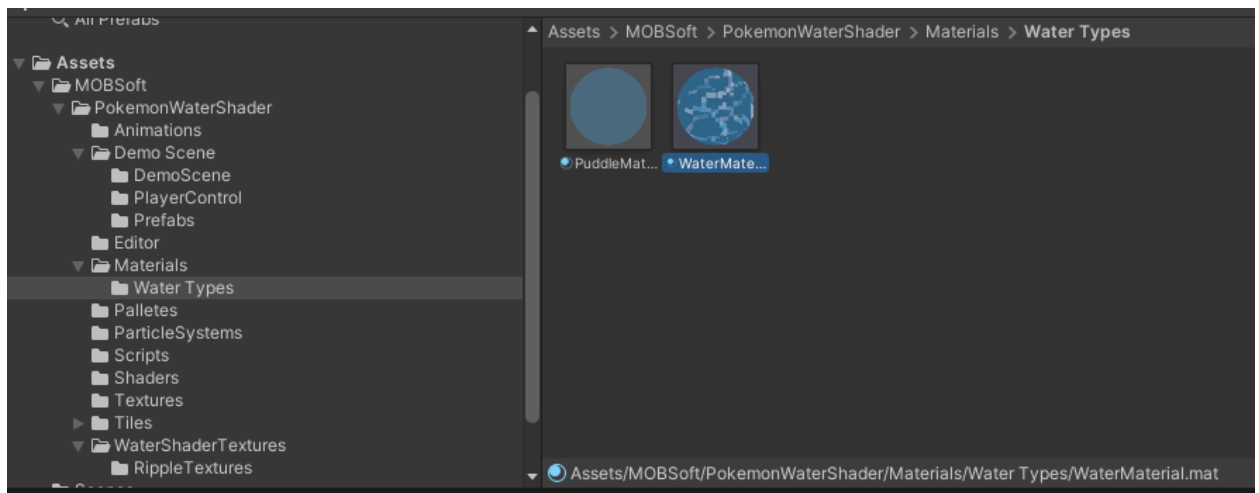
Using the Water Shader Material

Locating the Material
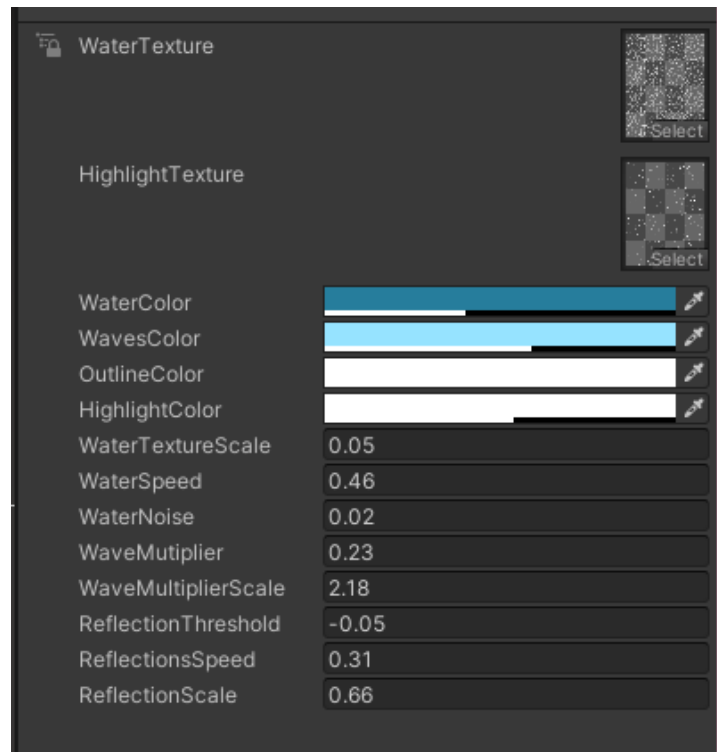- The water shader material ready to use is located in `MOBSoft > PixelStyleWaterShader > Materials > WaterType.`

Material Components
- This material comes with a base texture and a highlight texture provided for your convenience. You can use these textures as they are or replace them with your own.

Customization Properties

- **Water Color**: Adjusts the base color of the material.
- **Waves Color**: Modifies the refraction texture color.
- **Outline Color**: Changes the color of the shoreline outline.
- **Highlight Color**: Sets the color of light reflections on the water.
- **Water Texture Scale**: Scales the size of all textures (WaterTexture and HighlightTexture).
- **Water Speed**: Controls the speed of the movement of the WaterTexture.
- **Water Noise:** Alters the distortion level of the water.
- **Wave Multiplier**: Adjusts the visibility of the base color over the highlight.
- **Wave Multiplier Scale**: Changes the size of the base color texture.
- **Reflection Threshold**: Modifies the visibility of highlights on top of the waves.
- **Reflection Speed**: Controls the speed of the movement of the HighlightTexture.
- **Reflection Scale**: Adjusts the size of the HighlightTexture.

Applying the Material

- This material can be added like any other material on sprites *(It's important to remember that the sprite needs to be black to render the texture correctly)* or used as a Tilemap material.
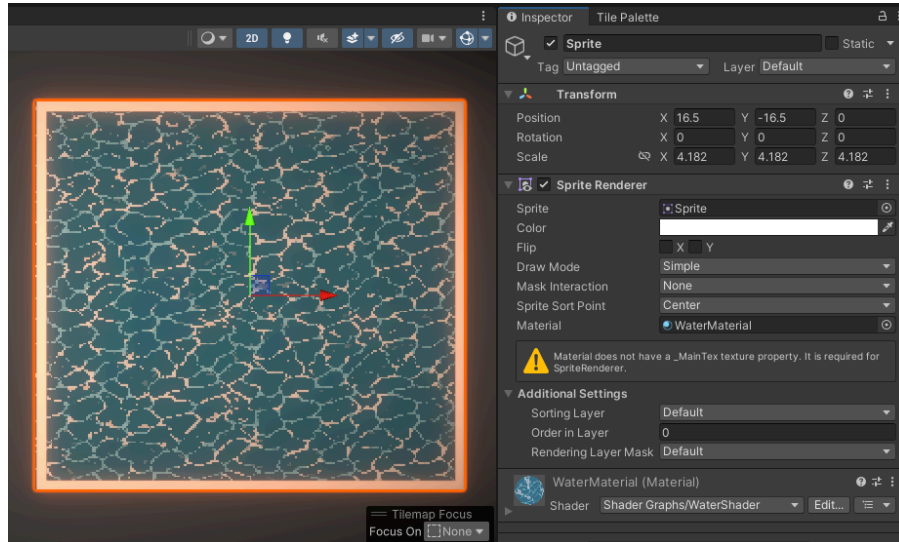

Setting Up Tilemap Rendering

- Creating a Tilemap Mask: To use the material with a Tilemap, create a mask of the area you want to render the shader on. The mask should be black with a white outline, proportional to the desired shoreline size.
- Tilemap Assets: Find necessary assets for ground and water Tilemaps in `MOBSoft > PixelStyleWaterShader > Textures.`
- Assigning the Material: Assign the Water Material to the mask's Tilemap.
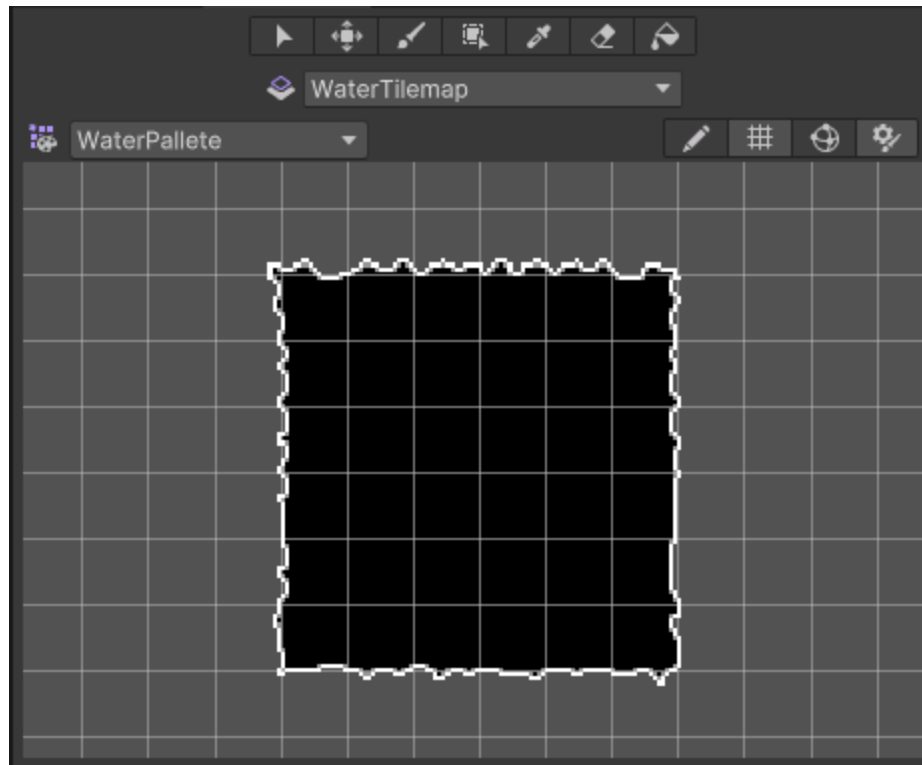
Recommended Setup for Tilemap:

- Attach the `TilemapMask.cs` script (located in `MOBSoft > PixelStyleWaterShader > Scripts`) to the Tilemap.
- Create an empty prefab containing a Sprite Mask component.
- Use the script to automatically generate a mask of your Tilemap as a child object, enabling reflections, foam effects, and ripples.
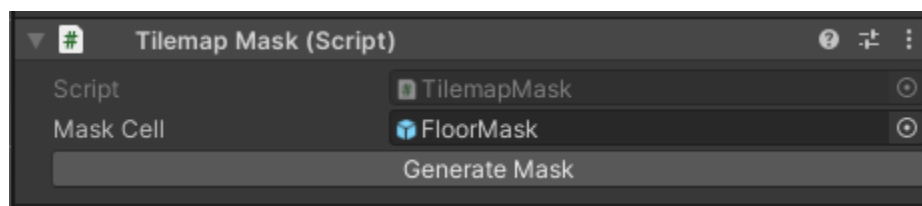
Implementing Visual Effects

Reflections

Assigning the WaterReflectableScript
- To enable reflections, assign the `WaterReflectableScript.cs` (located in `MOBSoft` > **PixelStyleWaterShader** > `Scripts`) to the desired objects or characters.

Customizing Reflection Properties
- The `WaterReflectableScript` allows precise control over reflection properties:
    - Local Position: Adjusts the X, Y, and Z position of the object's reflection.
    - Local Rotation: Changes the X, Y, and Z rotation of the object's reflection.
    - Sprite: This field should remain empty as it will automatically reference the object's sprite.
    - Sprite Layer: Modifies the layer assigned to the reflection.
    - Sprite Material: Use the recommended "ObjectReflection" material (located in `MOBSoft` > **PixelStyleWaterShader** > `Materials`) for realistic water movement effects.
    - Mask Interaction: Set this property to "Visible Inside Mask" to ensure that reflections are rendered only within the water's mask area.
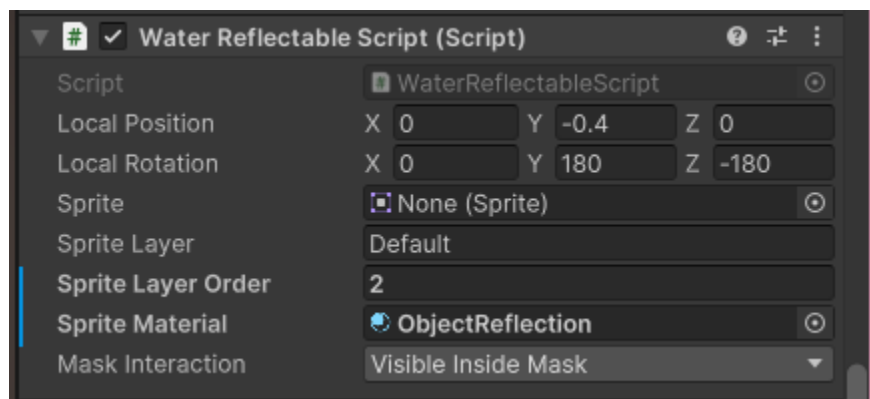
Foam Effects

Using the ObjectFoam Prefab
- Drag and drop the "ObjectFoam" prefab (found in `MOBSoft >`
  `PixelStyleWaterShader > ParticleSystems`) onto the desired objects.
- Adjust the size of the foam to fit the proportions of the object.
- Customize the foam effect using provided sprites such as `Ripple.png`,
  `Ripple2.png`, and `Ripple3.png` (located in `MOBSoft >`
  `PixelStyleWaterShader > WaterShaderTextures > RippleTextures`).

Ripple Effects

Using the FootstepFXManager
- Add the `FootstepFXManager` prefab (located in `MOBSoft >`
  `PixelStyleWaterShader > DemoScene > Prefabs`) to your scene.
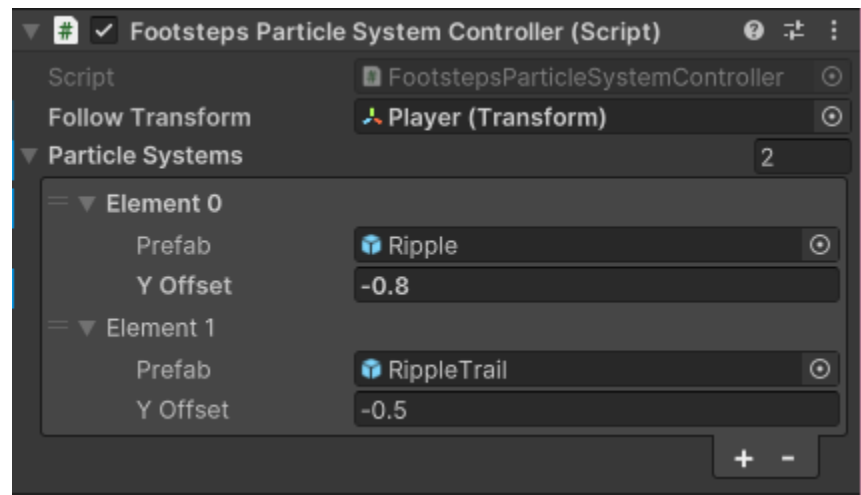
Configuring the FootstepFXManager
- The `FootstepFXManager` contains the
  `FootstepParticleSystemController` script (found in `MOBSoft >`
  `PixelStyleWaterShader > Scripts >`
  `FootstepsParticleSystemController.cs`).

Modifying Particle Systems

- Configure the following properties within the `FootstepFXManager`:
  - Follow Transform: Assign the object that both the Ripple and Ripple Trails should follow.
  - Particle Systems: Choose between "Ripple" and "RippleTrail" particle systems:
    - "Ripple": Creates animated ripples emitted from characters while sitting on a water surface.
    - "RippleTrail": Generates a trail of ripples that follow characters and disappear after a specified duration.
- Both particle systems (`Ripple` and `RippleTrail`) can be customized or replaced and are located in `MOBSoft > PixelStyleWaterShader > ParticleSystems`.

Environmental Visual Effects

Clouds

Using the Clouds Prefab
- Locate the "Clouds" prefab under `MOBSoft > PixelStyleWaterShader > DemoScene > Prefabs`.
- This prefab is ready to use for cloud effects.

Configuring the Clouds Prefab
- The "Clouds" prefab contains a Sprite Renderer with an animator that simulates clouds moving across the sky.
- Customize the appearance of the clouds by replacing the provided "Clouds" texture (found in `MOBSoft > PixelStyleWaterShader > Textures`) with your own texture if desired.

Setting Mask Interaction

- The prefab is already configured with a Mask Interaction of "Visible Inside Mask", ensuring that the clouds are only visible within the water area defined by the mask.

Layer Order

- Ensure that the Clouds VFX is in the same Order in Layer as your water's TileMap for proper rendering.

Rain

Using the Rain Prefab

- Find the "Rain" prefab under `MOBSoft > PixelStyleWaterShader > DemoScene > Prefabs.`
- This prefab provides rain visual effects.

Customizing Rain Parameters

- The "Rain" Particle System includes a main rain effect and two child objects named "Ripple".
- Adjust the intensity of the rain by modifying the "Rate over Time" parameter under the "Rain" Particle System > Emission settings.

Setting Rain Ripples

- Rain ripples are pre-configured to appear only inside the water area defined by the mask.