SIGGRAPH2010

Water Flow in PORTAL 2

Alex Vlachos, Valve

July 28, 2010

VALVE

Outline

- Goals & Technical Constraints
- How Artists Create Flow Maps
- Flowing Normal Maps in Left 4 Dead 2
- Flowing Color Maps in Portal 2


Rochelle
Coach
Ellis
+100

Goals

- Visual
  - Solve repeating texture artifacts
  - Flow around obstacles
  - Vary water speed and bump strength
- Technical
  - Work with existing reflective surfaces
  - Min hardware ps2.0b (6-year-old hardware) & Xbox 360
- Gameplay…

Gameplay

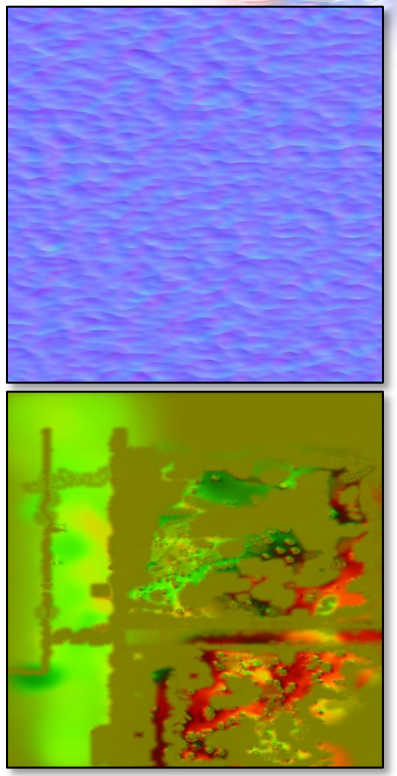
- Early Left 4 Dead 2 playtests showed players were confused and got lost often in the swamps
  - Soft non-directional lighting
  - Trees provided too much cover
- My theory was that water flow would improve gameplay by highlighting the correct path
- We tested this theory through playtesting
- In practice, we found testers took 17% fewer wrong turns and decreased the time it took to traverse the level!

Technical Constraints

- Already at perf limits on the Xbox 360 & low-end PC
- Already at memory limits on the Xbox 360
- Our water shader had limited instructions left for our low end hardware ps2.0b

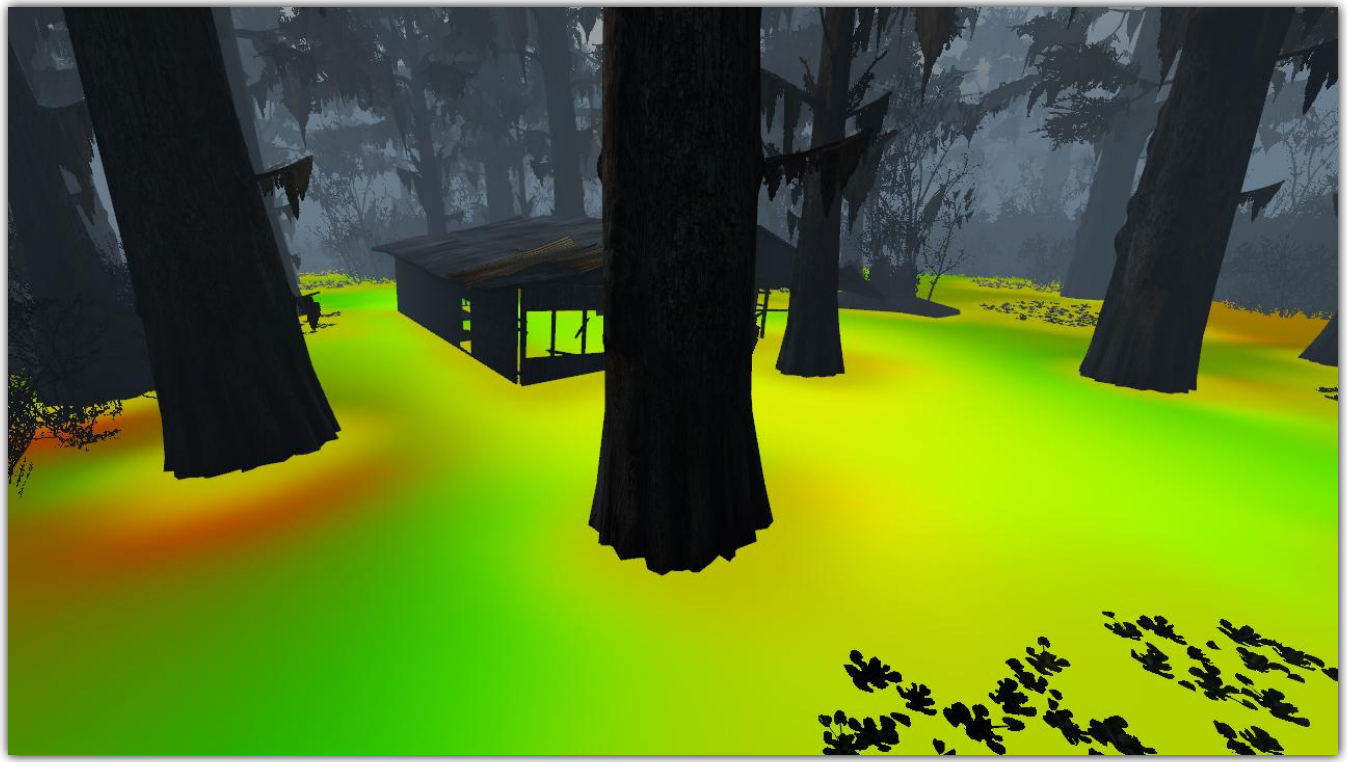
Algorithm Overview

- Pixel shader flow, not geometric flow
- Continue to use a normal map for water ripples
- Artists author a flow map (a texture containing 2D flow vectors)
- Use this flow map in a pixel shader to distort the normal map in the direction of flow

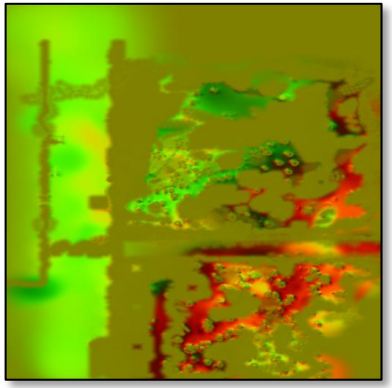


Flow Texture Mapped onto Surface

Covers entire water surface

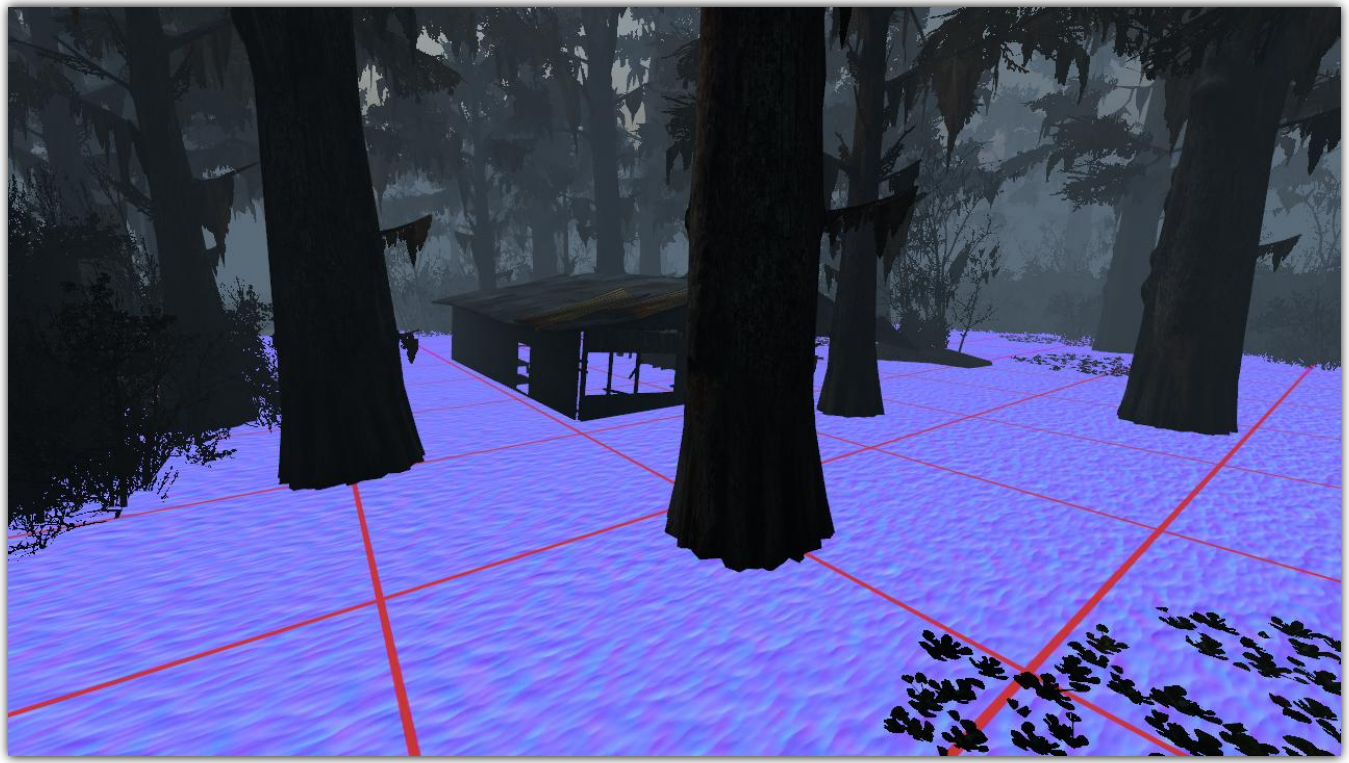


Flow Texture

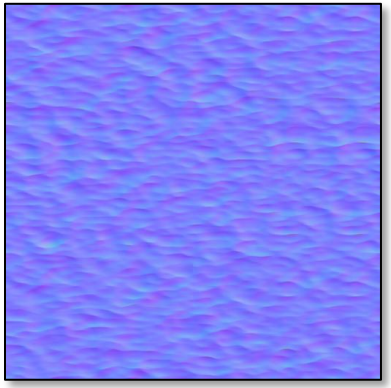


Normal Map Mapped onto Surface

Tiled over the water surface



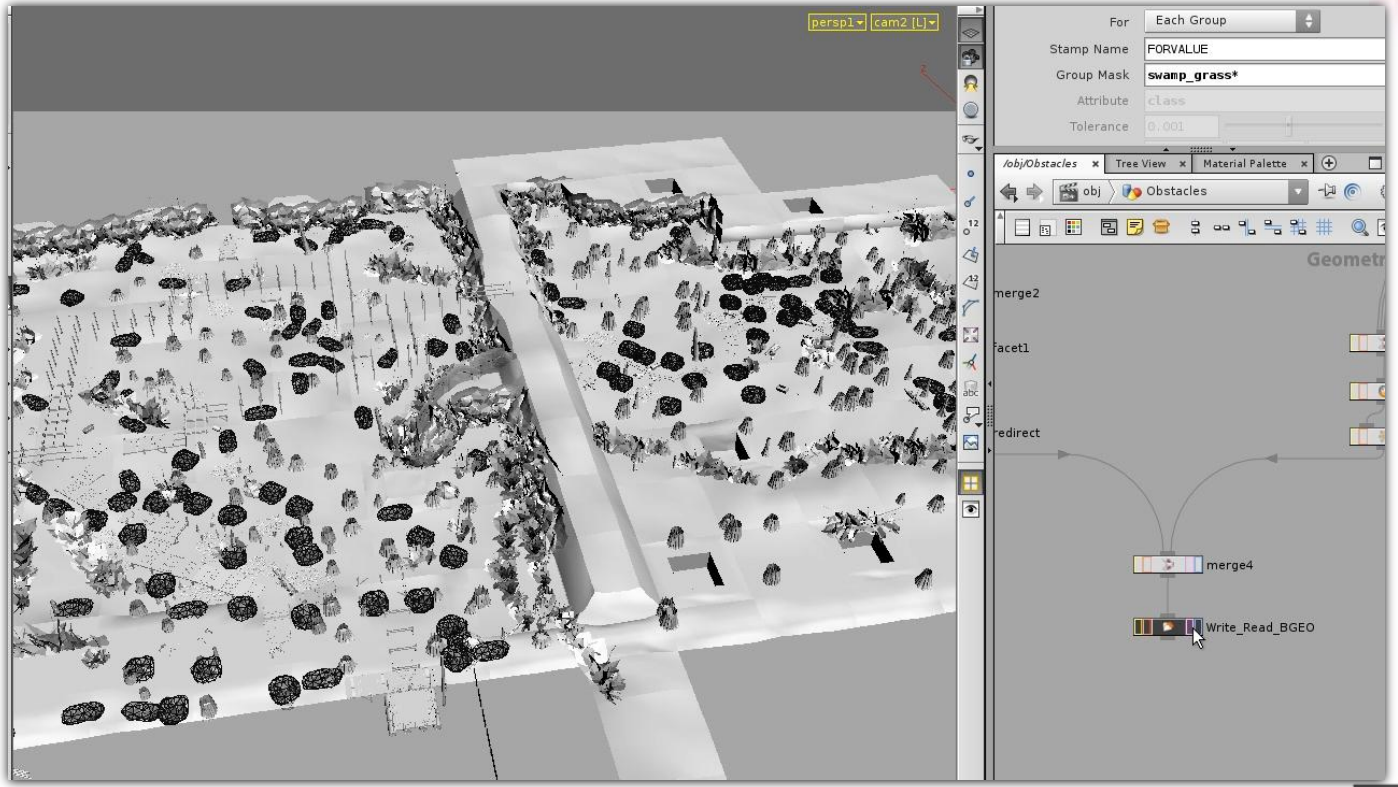
Normal Map



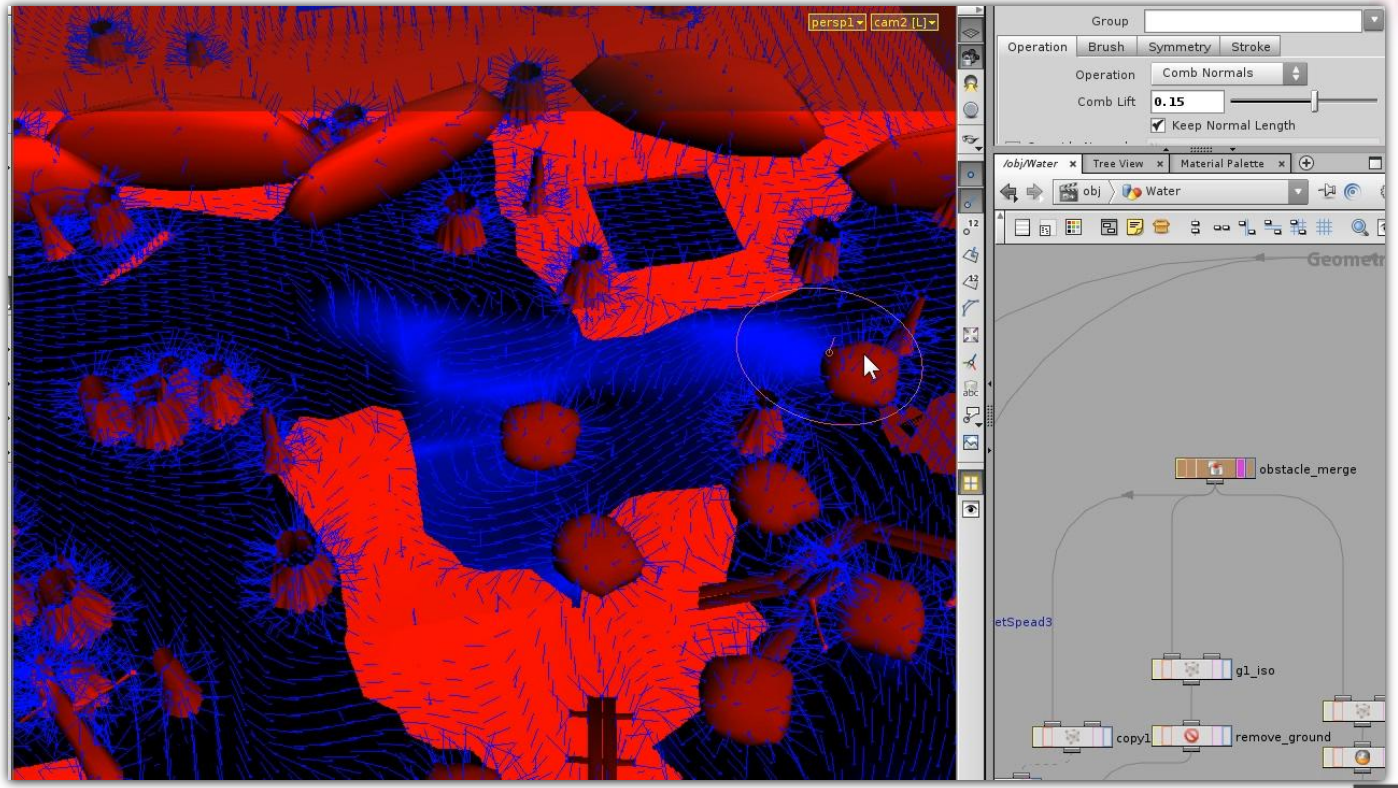
Artists Author Flow Maps

- Flow map provides a unique 2D vector for every point on the water surface
- Relatively low resolution: ~4 texels/meter
- Impractical to paint directly
- We use Houdini to create vector flow maps

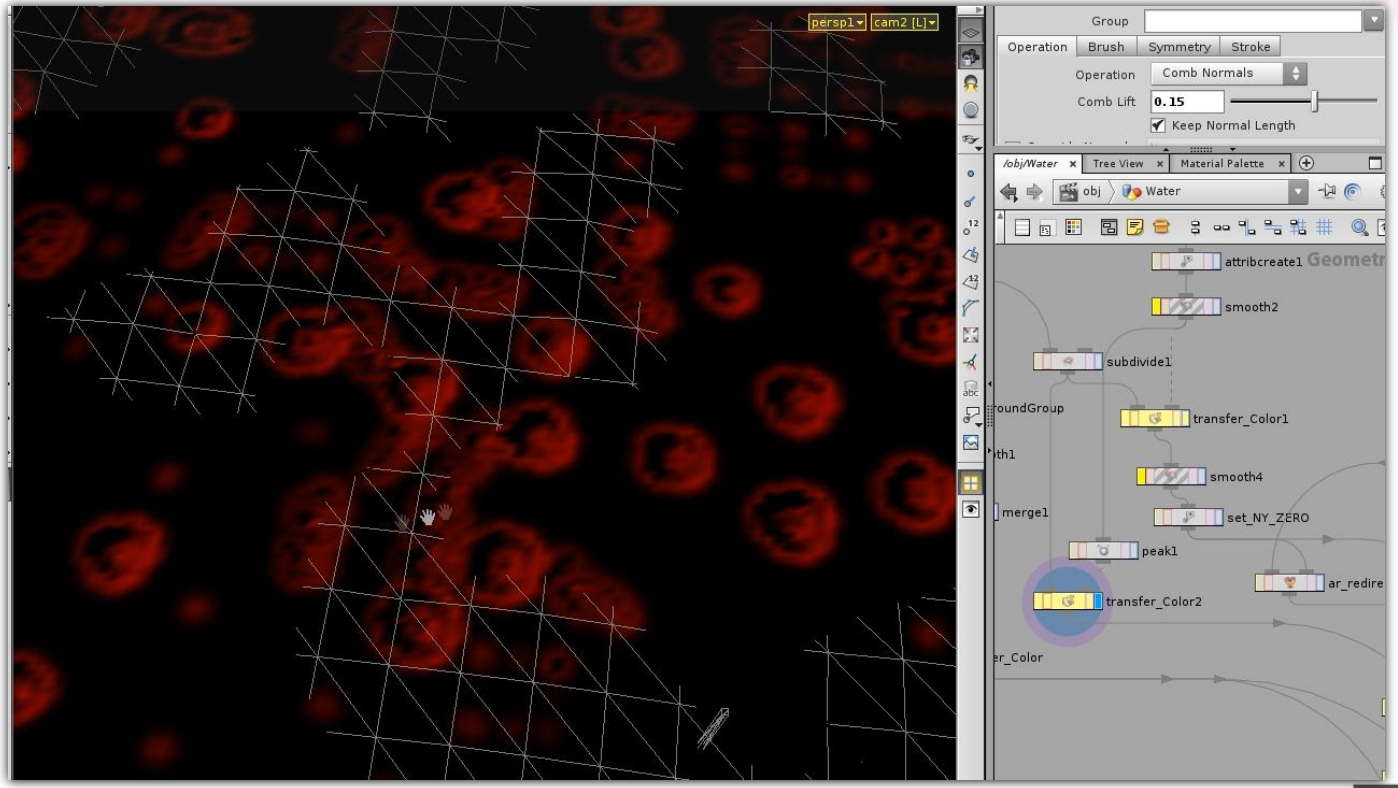
Houdini – Importing Level Geometry



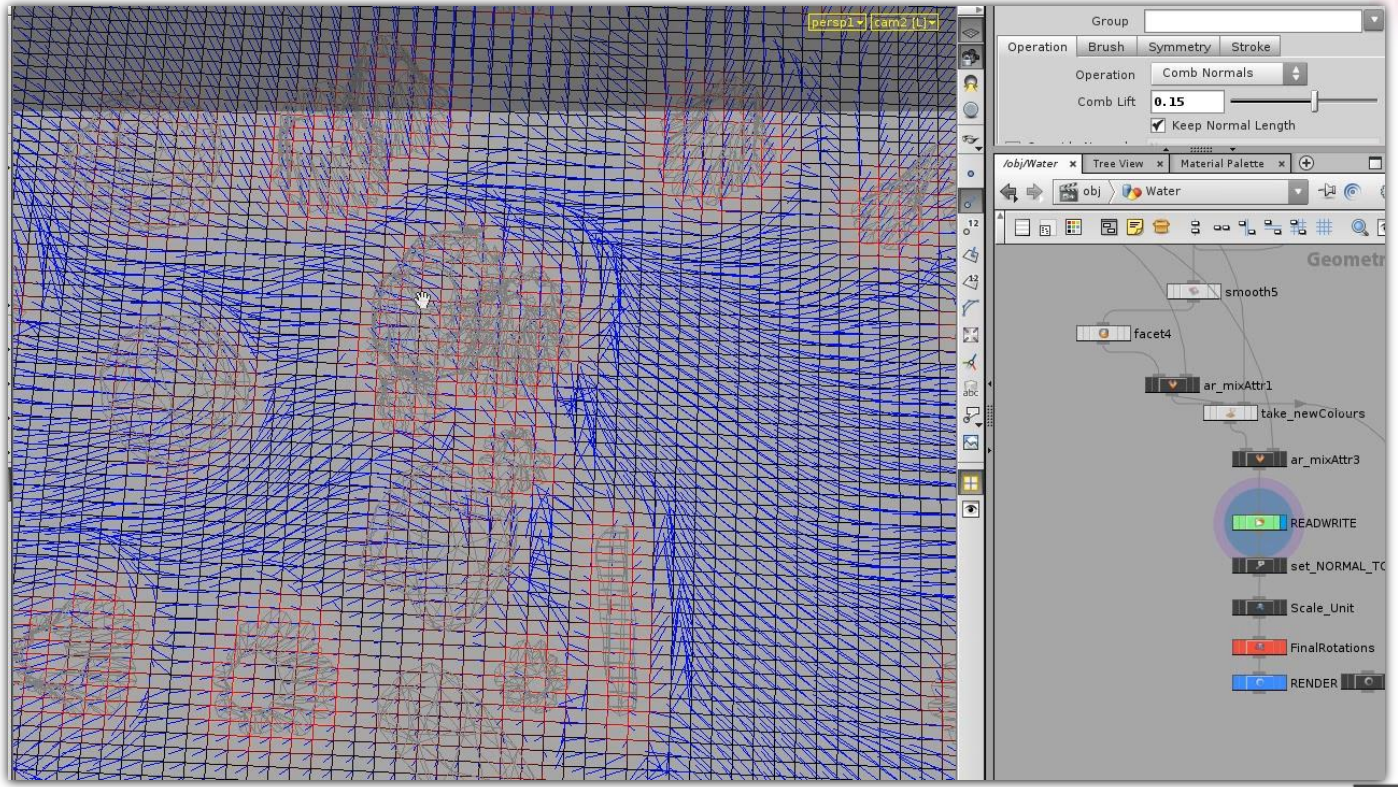
Houdini – “Combing” Vector Field



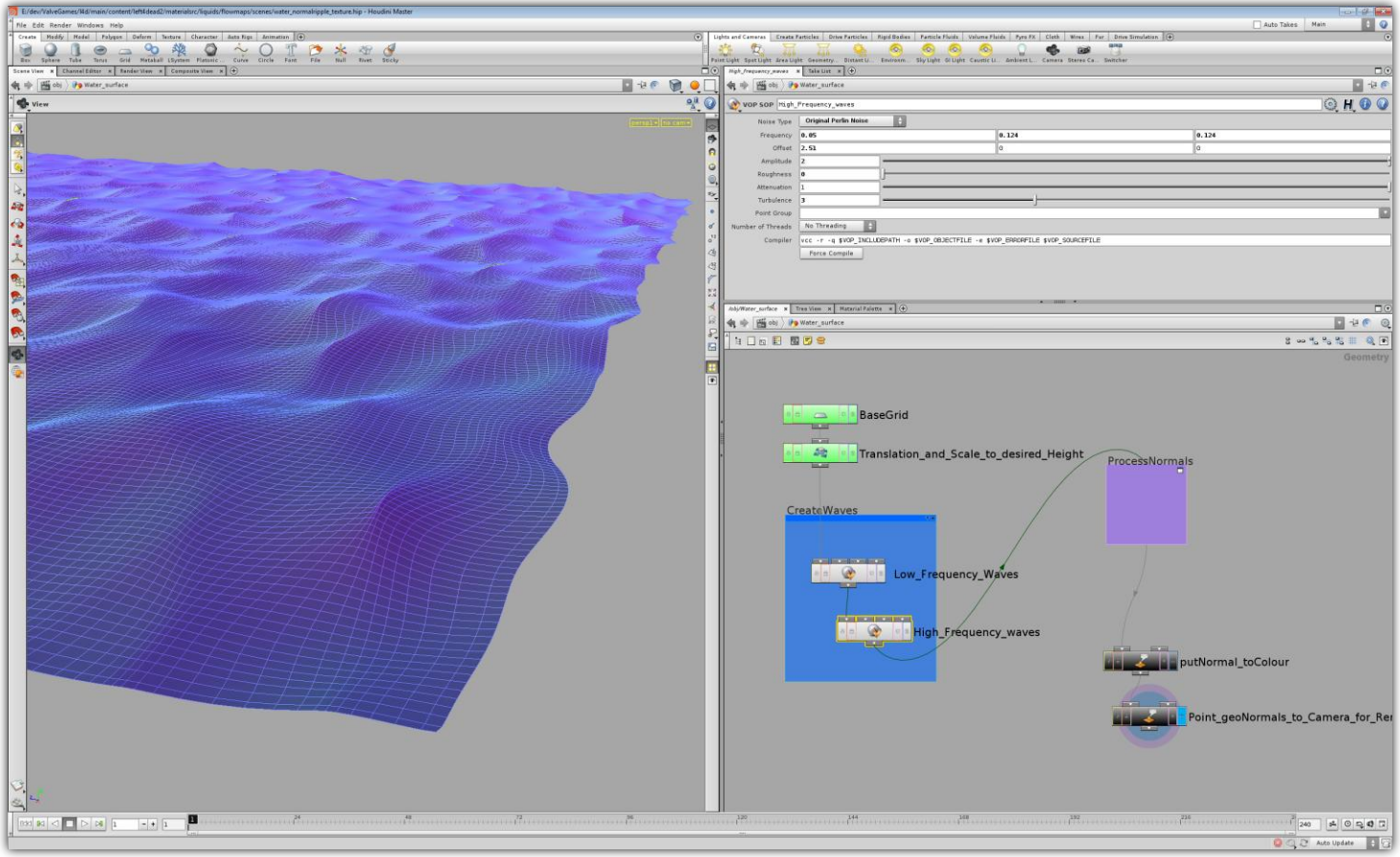
Houdini – Procedural Masks



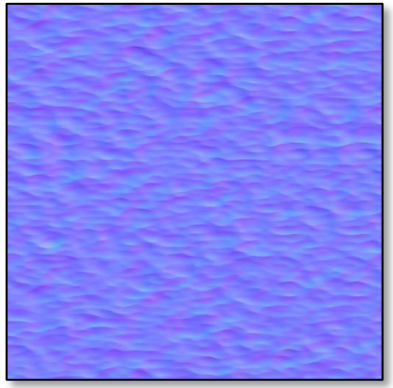
Houdini – Applying Masks



Houdini – Water Normal Maps



Normal Map



Left 4 Dead 2

- Wanted to replace our scrolling normal maps with flowing normal maps

- Keep the rest of the water shader the same

- This algorithm ultimately provides a new per-pixel normal generated from the normal map and flow map

Related Work

- Nelson Max and Barry Becker 1995. *Flow visualization using moving textures.* In Proceedings of the ICASW/LaRC Symposium on Visualizing Time-Varying Data, 77–87.

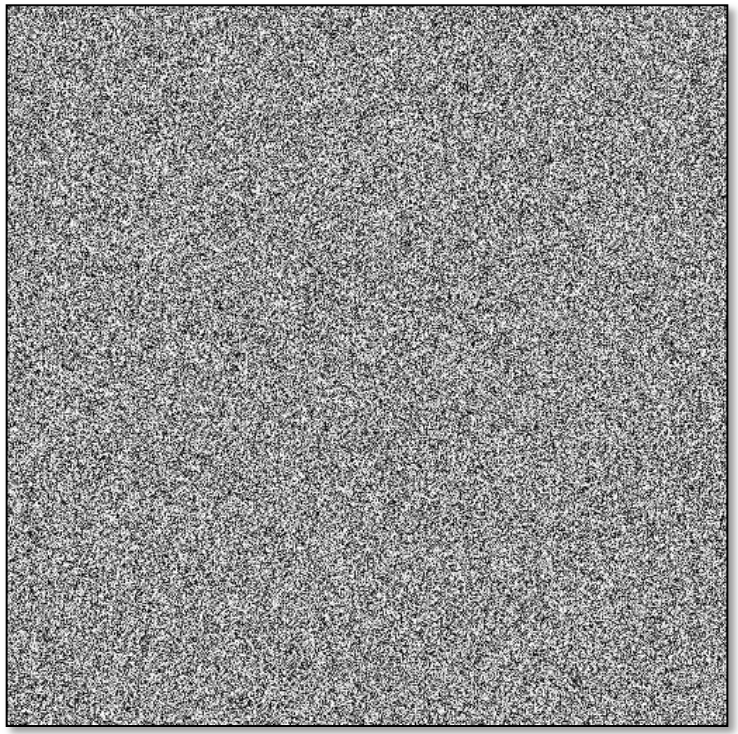
- Building on aspects of their algorithm and applying their approach to flowing normal maps

Flow Visualization

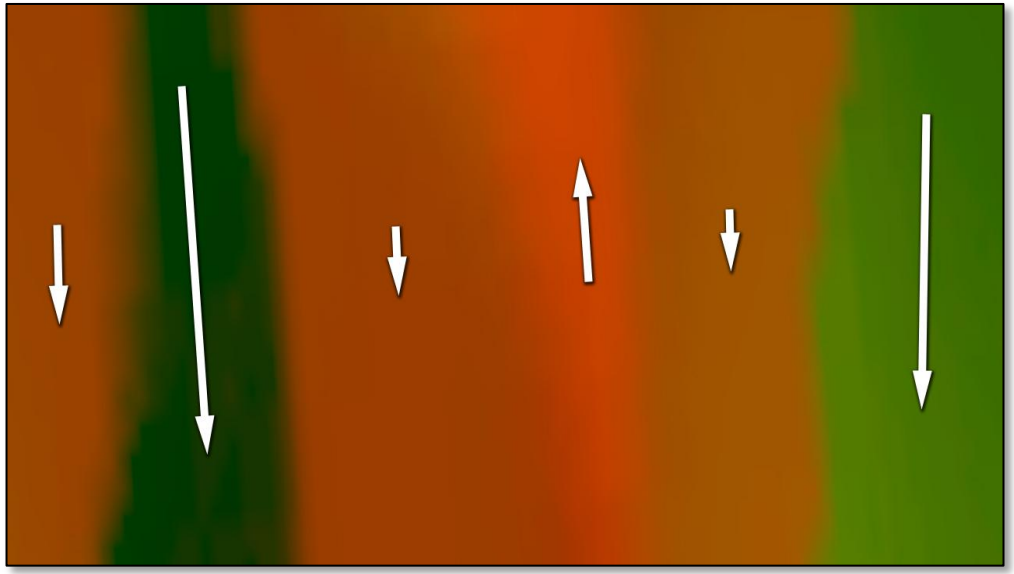
- Inputs: flow field & noise texture
- Distort a noise texture to visualize a flow field
- The UV is offset by the 2D flow vector scaled by time

Flow Visualization Textures

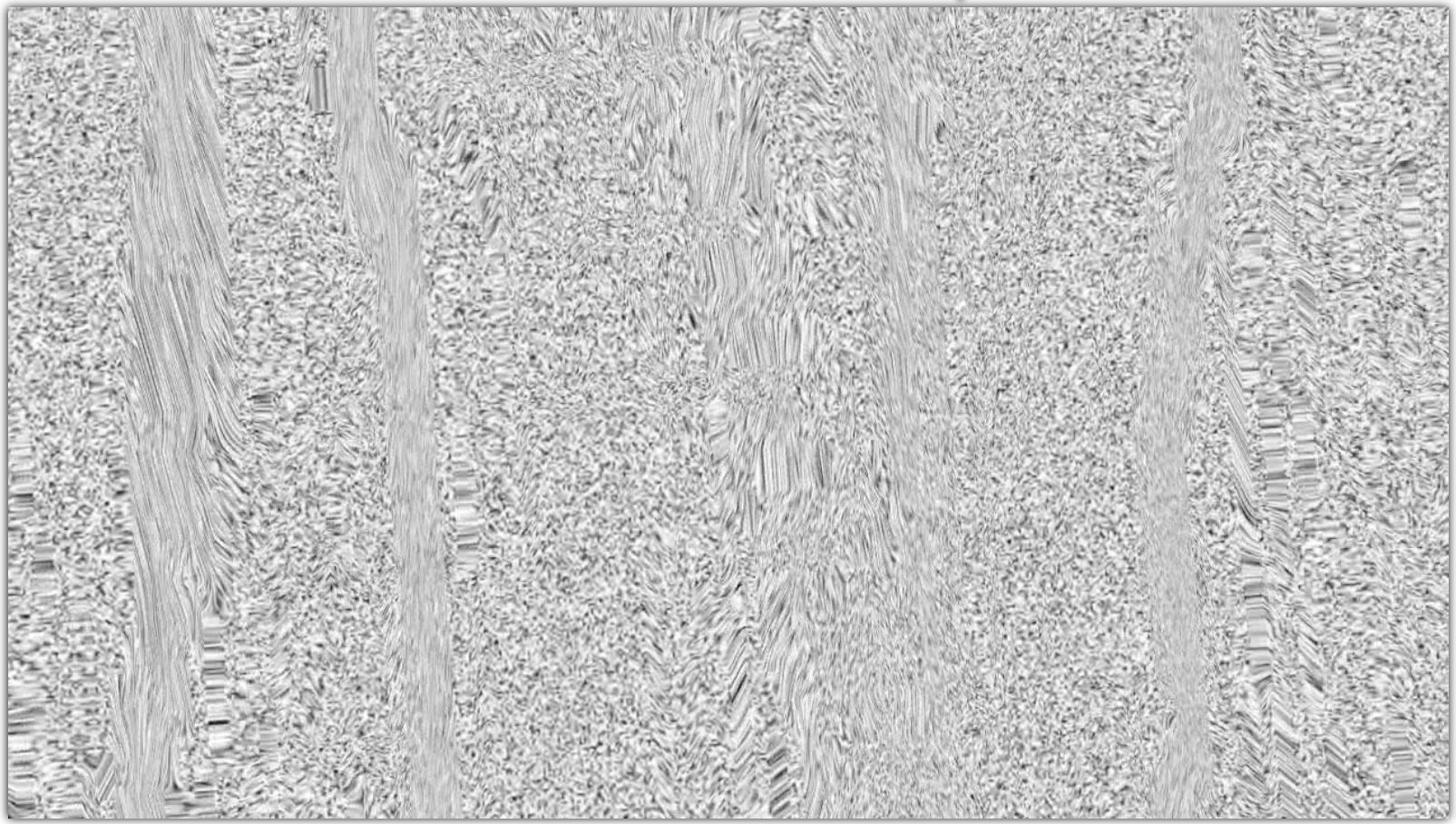
Noise Texture



Flow Texture



Flow Visualization Experiment



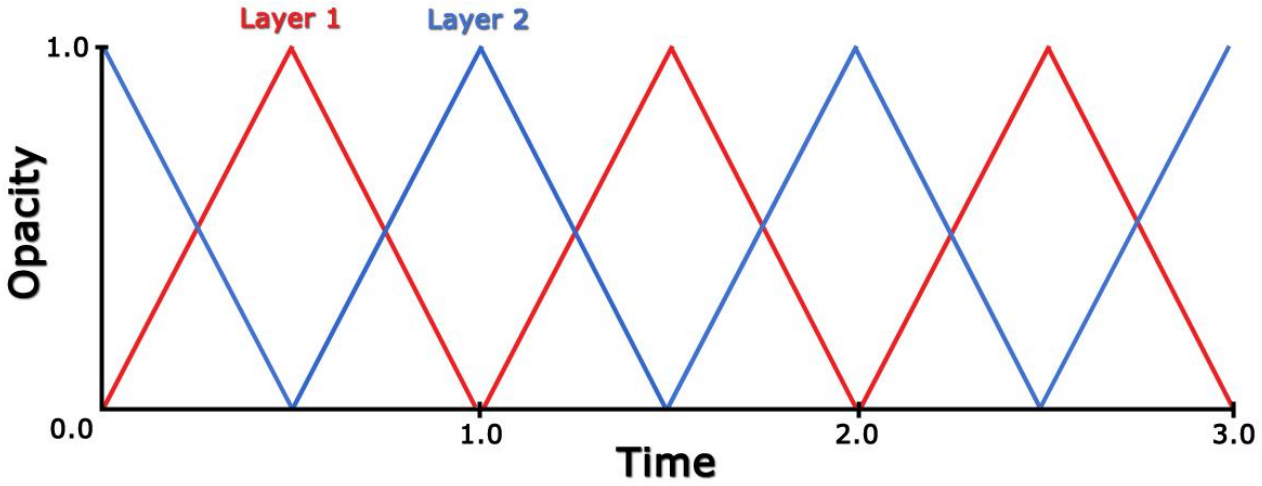
Flow Texture

Max & Becker's Observation

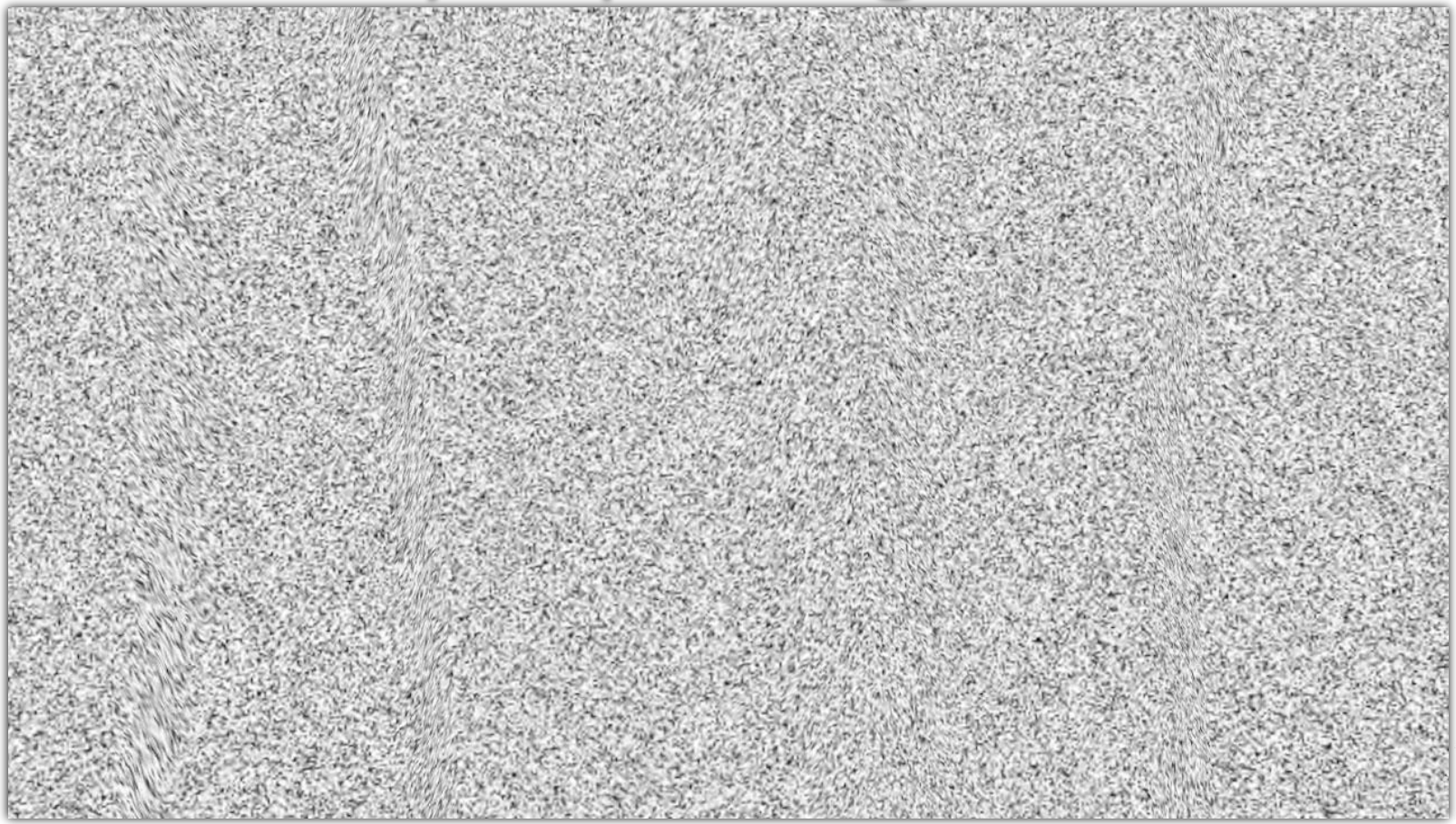
- The beginning of the distortion looks convincing
- Only distort a small amount
- In general, distortion looks reasonable for the first 1/3 of uv space

Smoothly Interpolating Layers

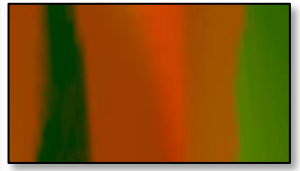
- Blend the short animated segment in two layers
- Each layer is offset half a phase so we can hide the restart for each layer



Smoothly Repeating Flow



Flow Texture



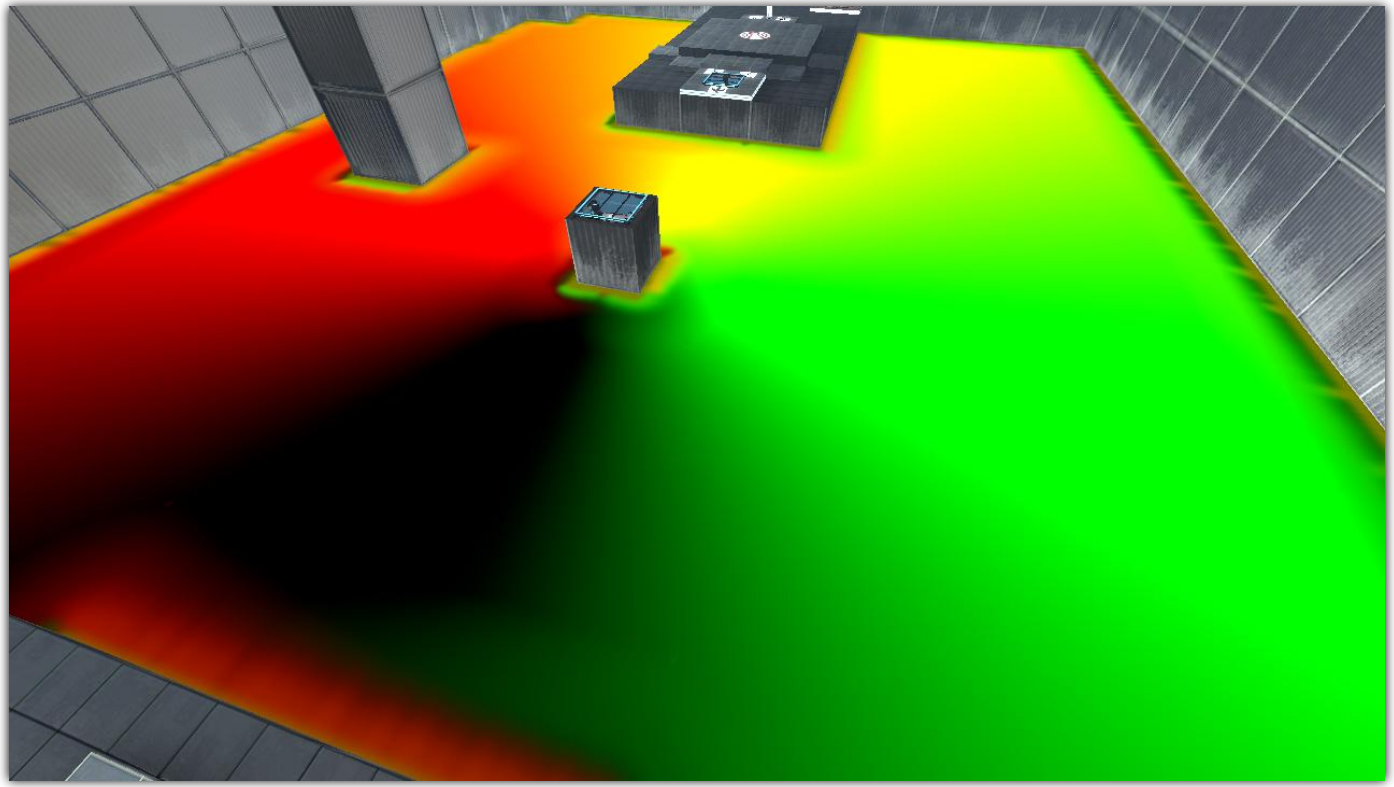
A Great Start

- We now have a method to flow a normal map
- We want to apply this to a larger surface
- But applying this to a large surface means tiling our normal map which will cause artifacts…

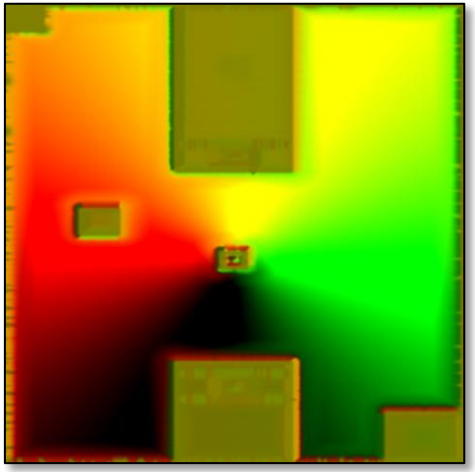
Portal 2 Test Map

Portal 2 Test Map (Programmer Art)

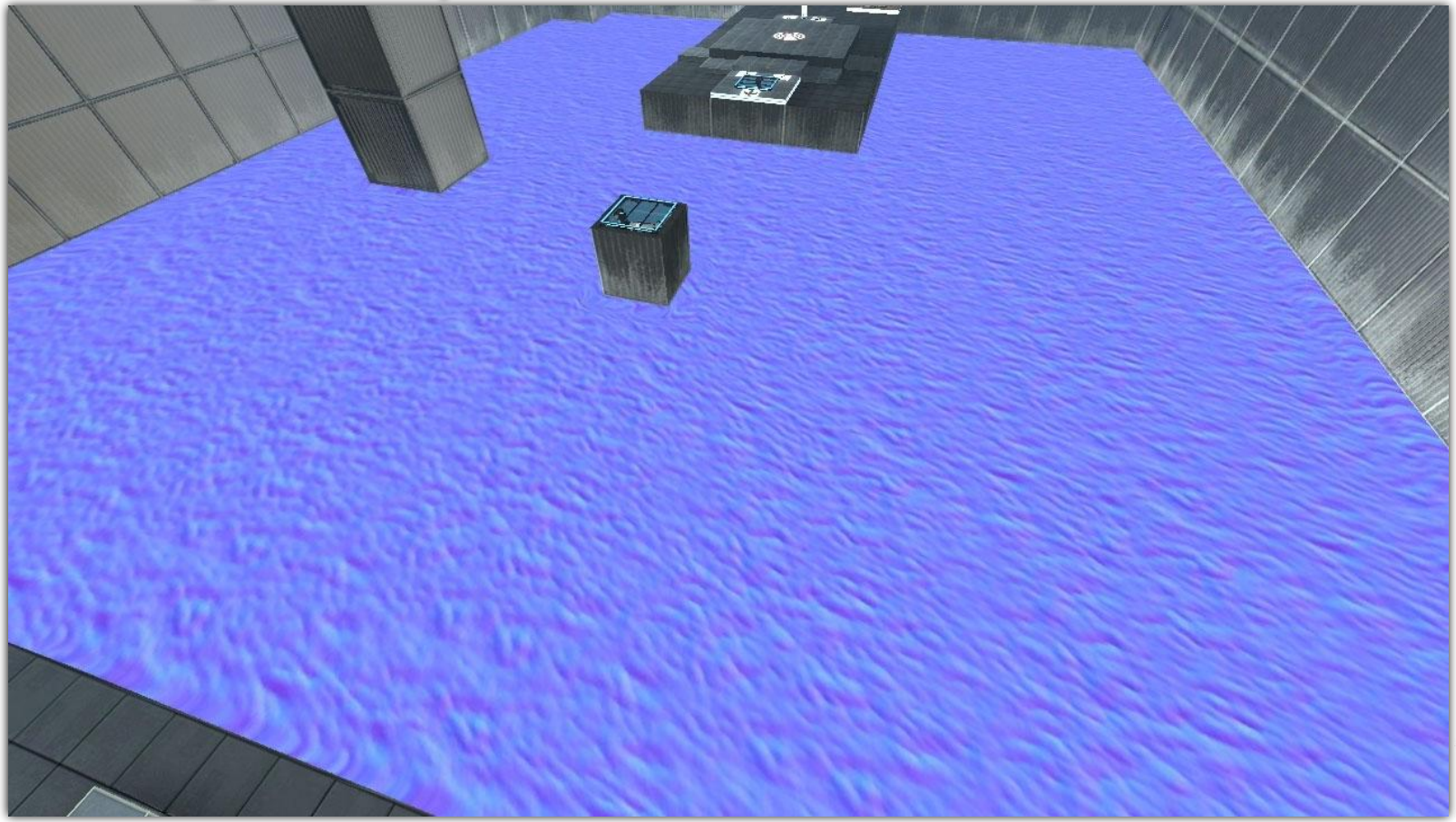
Flow Vectors on Water Surface



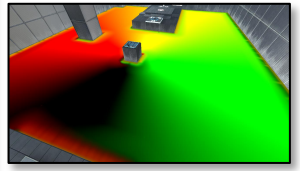
Flow Texture



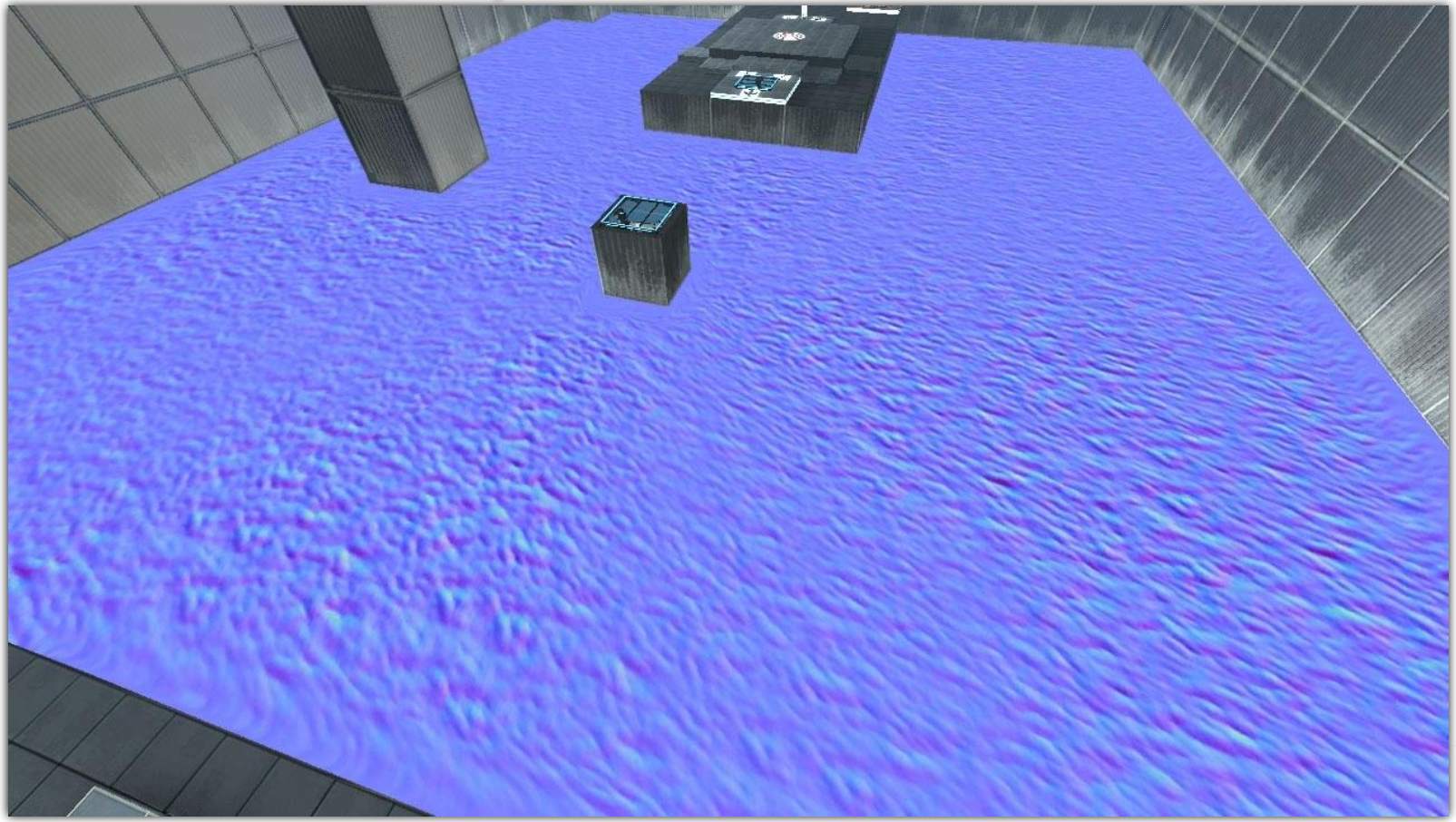
Single Layer Normal Distortion



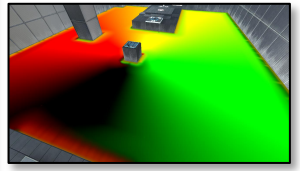
Flow Vectors



Double Layer Normal Distortion



Flow Vectors

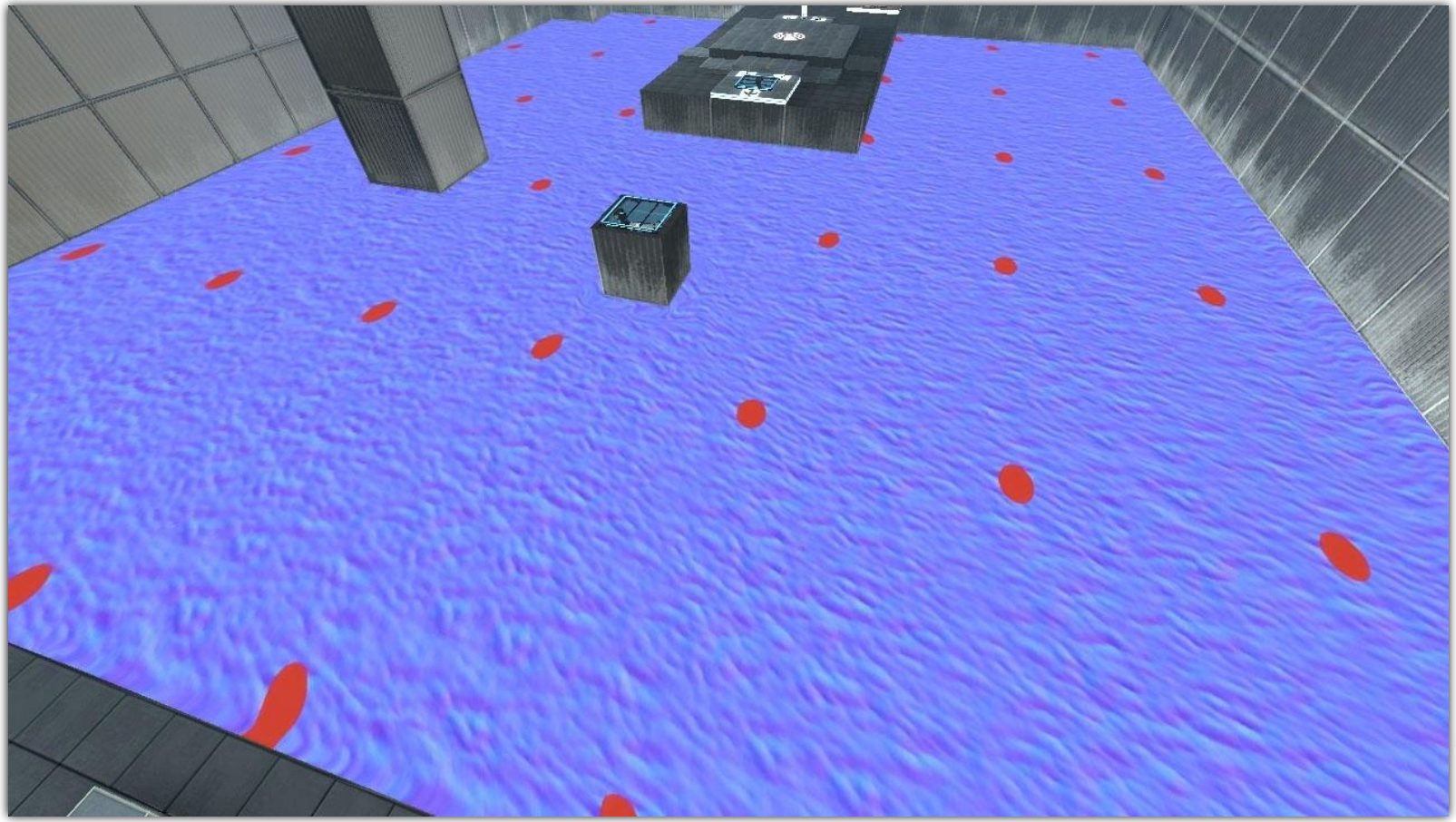


Two Major Problems

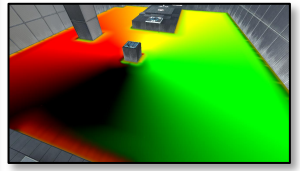
- Repetition – The same normals will flow through the same point on the mesh

- Pulsing – The surface appears to pulse in a repeating pattern

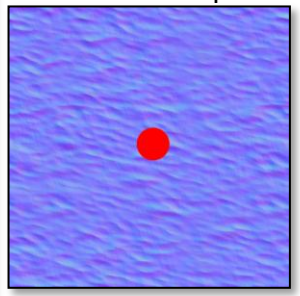
Repetition Visualization Single Layer



Flow Vectors

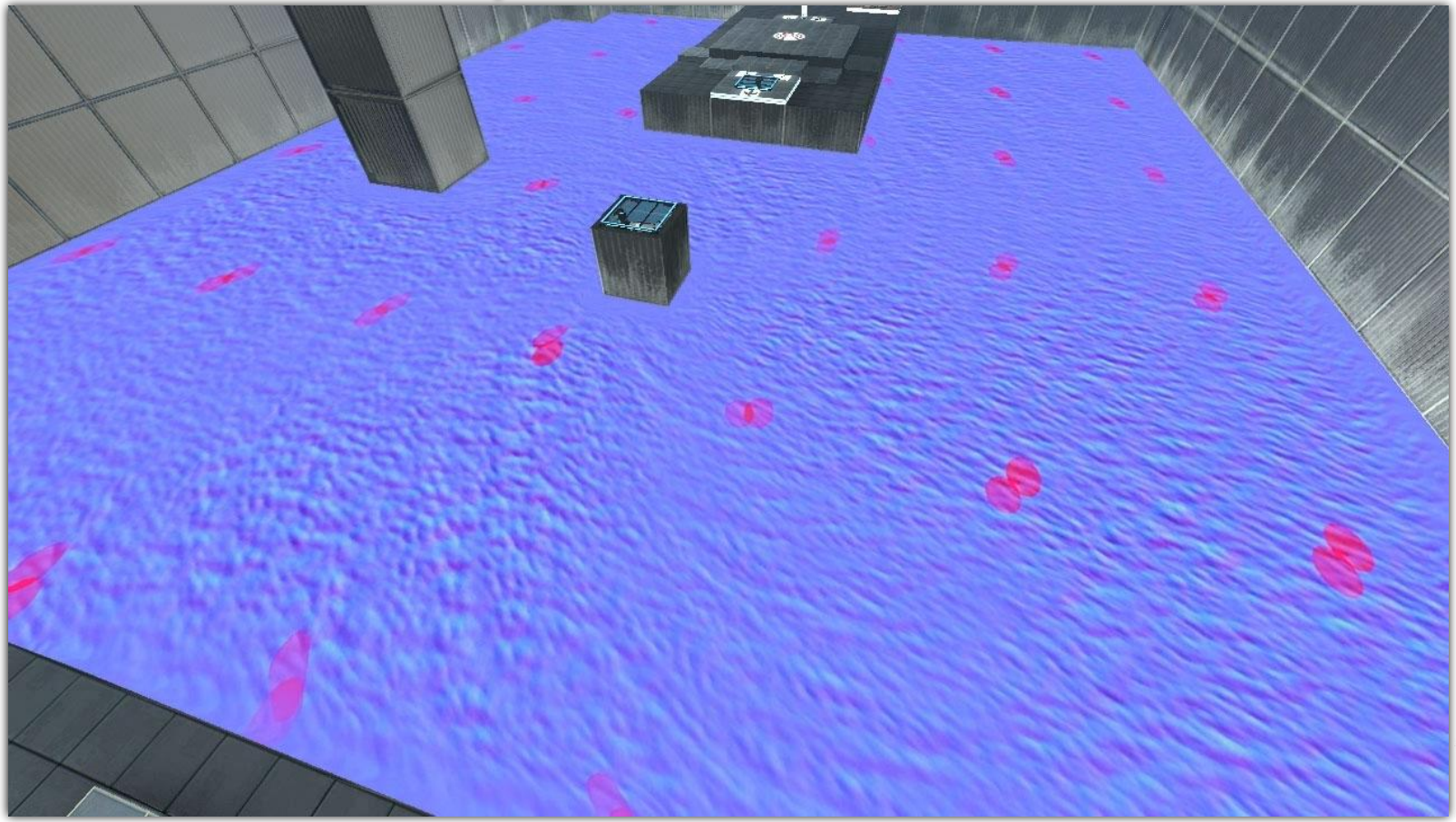


Normal Map

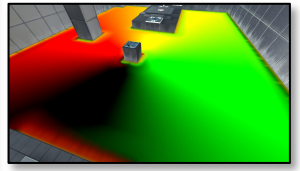


VALVE

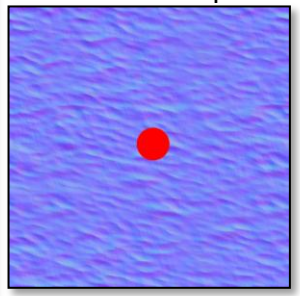
Double Layer



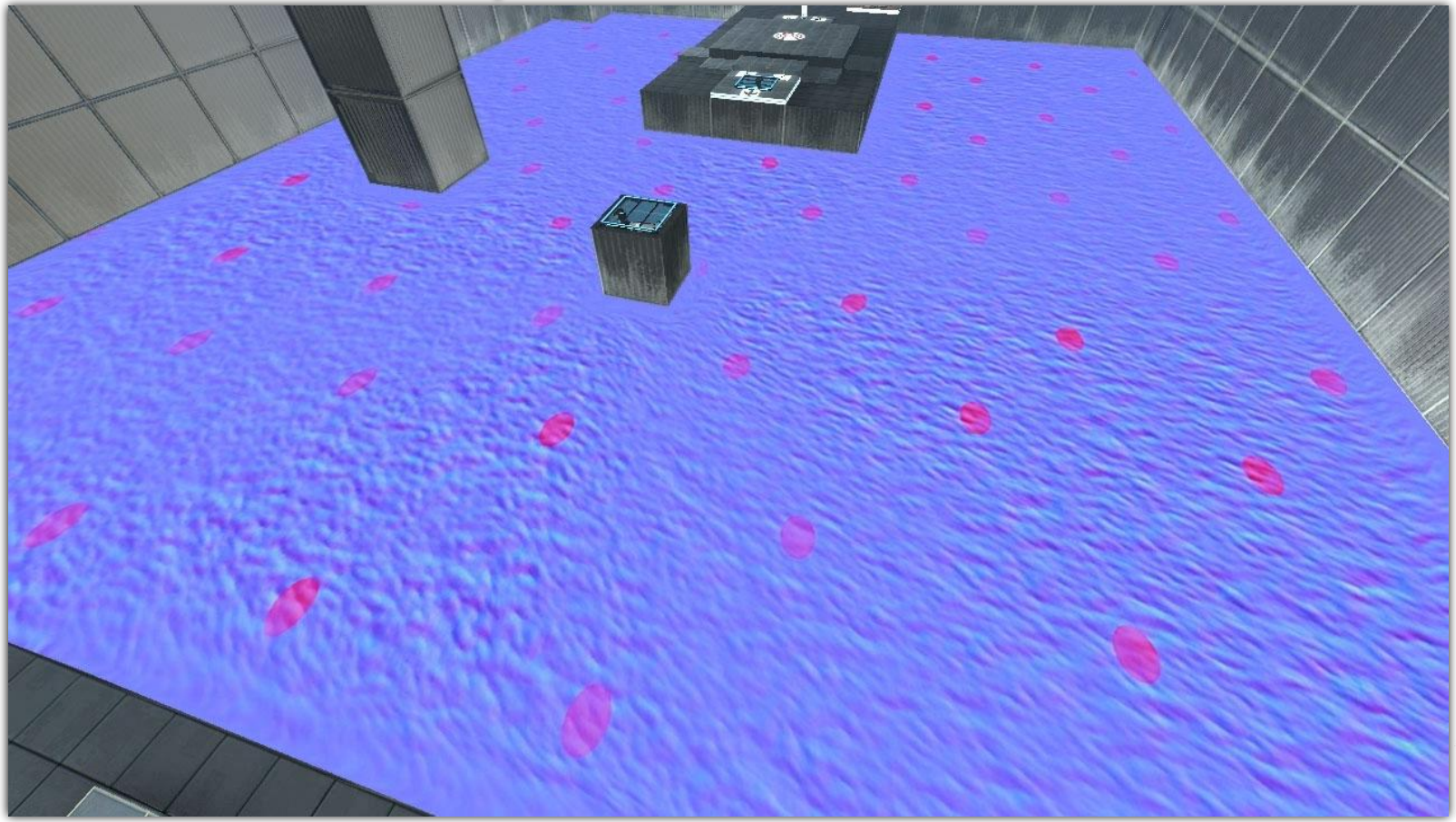
Flow Vectors



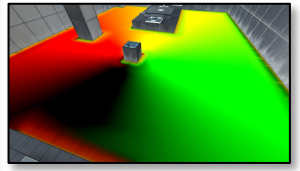
Normal Map



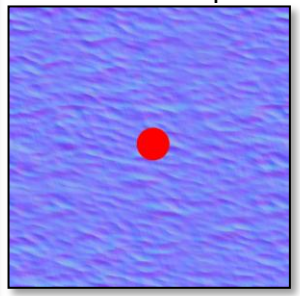
Double Layer With Offset



Flow Vectors

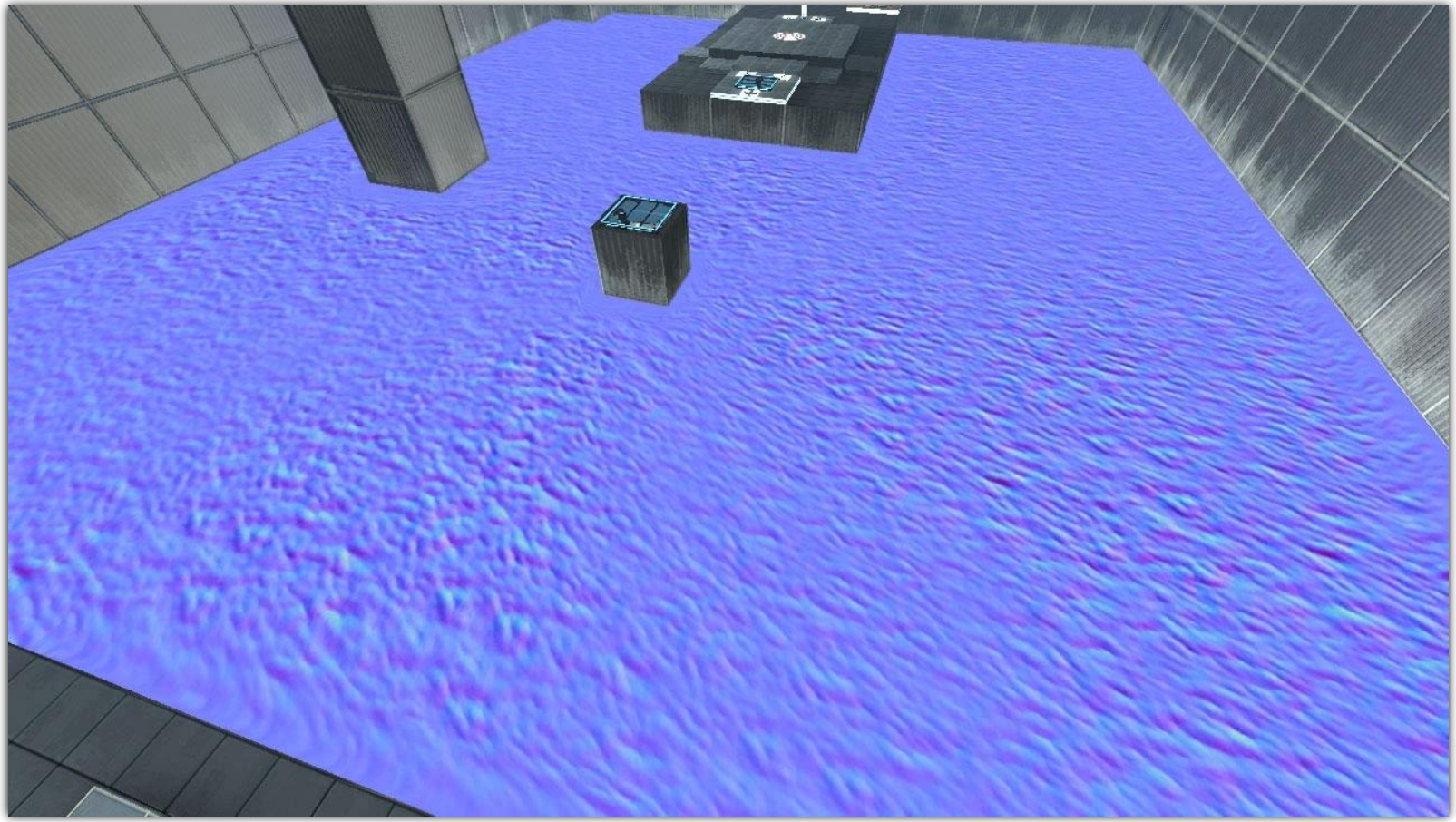


Normal Map

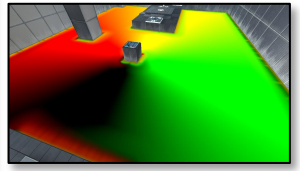


VALVE

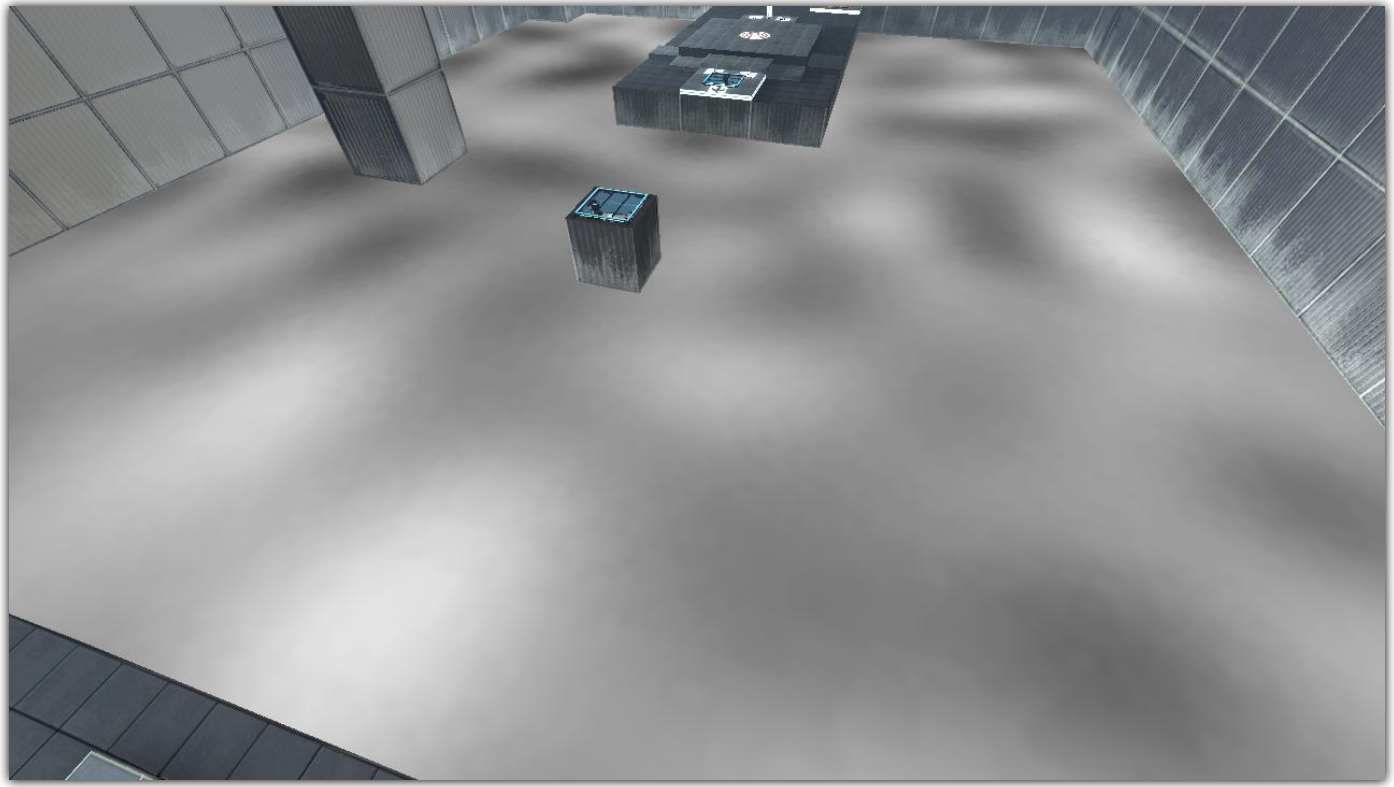
Repetition Solved by Offset



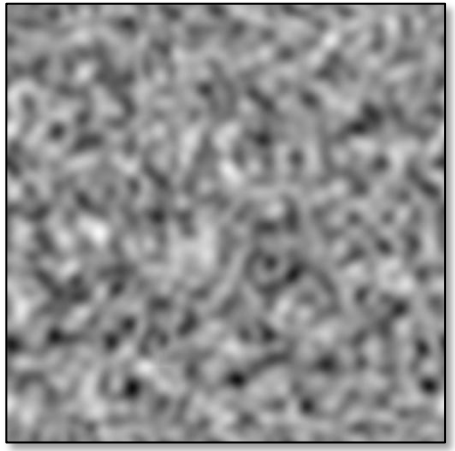
Flow Vectors



Pulsing Solved by Noise

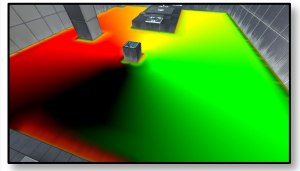


Noise Texture

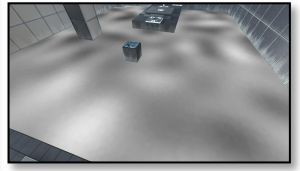


Pulsing Solved by Noise

Flow Vectors

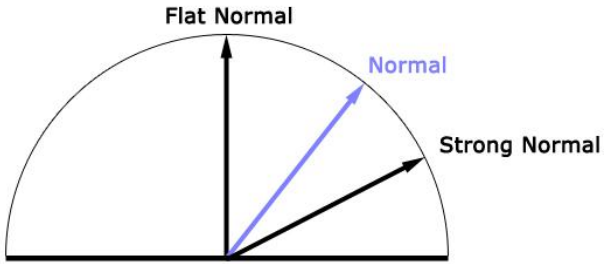


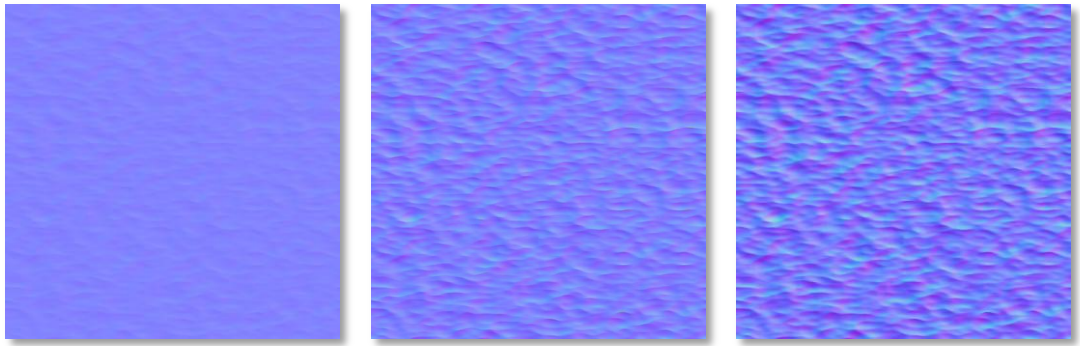
Noise Texture



Water Speed Affects Normals

We scale down the strength of the normal in tangent space by the flow speed (Flow speed is the length of the 2D flow vector)





Performance

Compared to scrolling two normal maps:

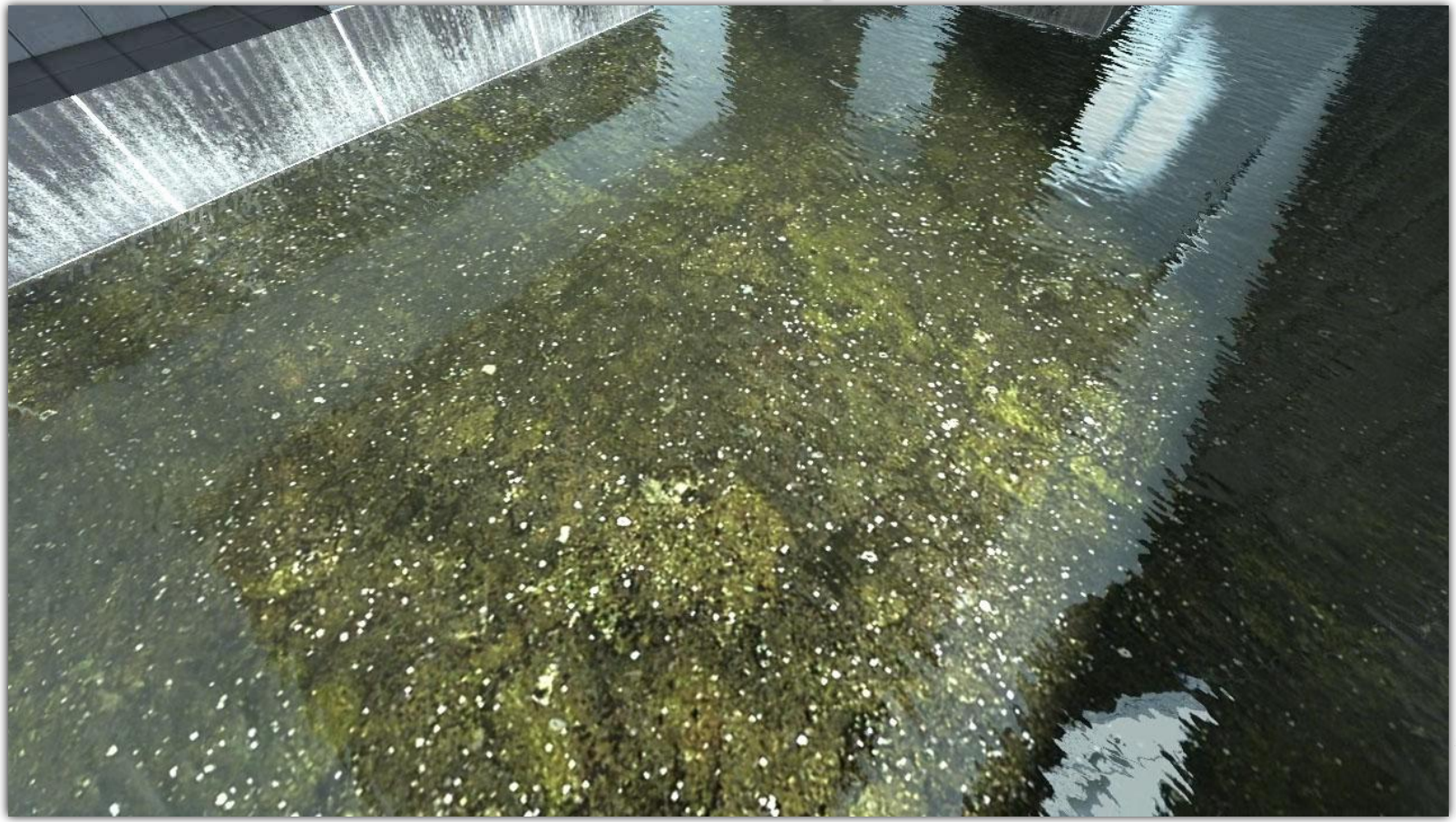
- Additional texture fetches: 2 - flow & noise
- Additional arithmetic pixel shader instructions: 21

Water Flow in Portal 2

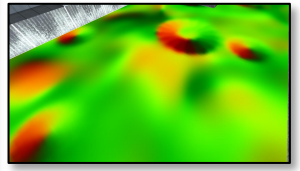
- Wanted to also flow debris in dirty water

- Needed to modify our algorithm to support flowing a color map

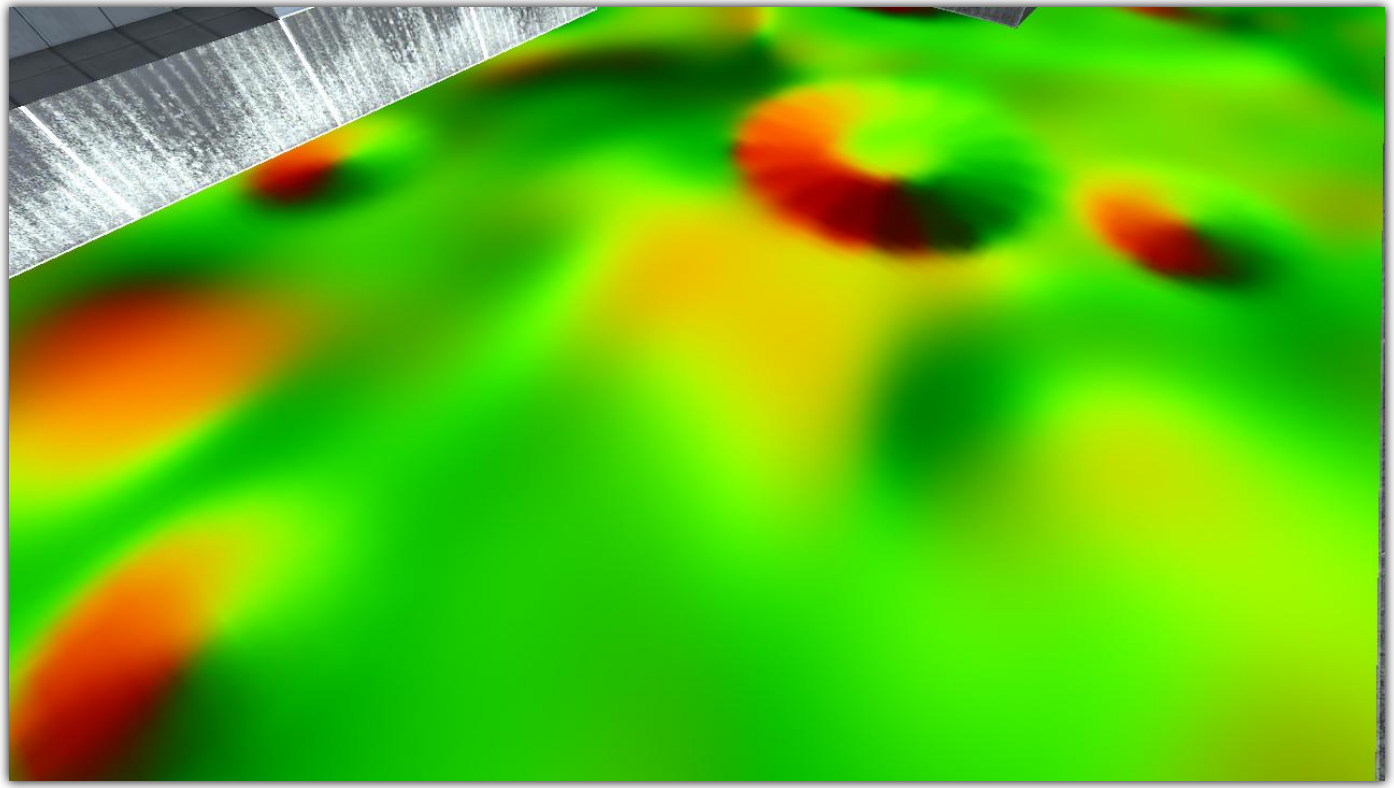
Debris Flow Example



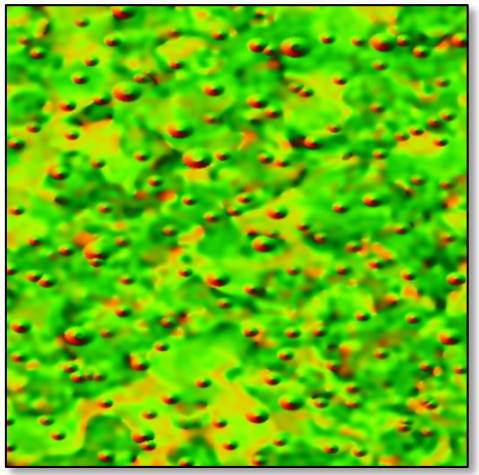
Flow Vectors



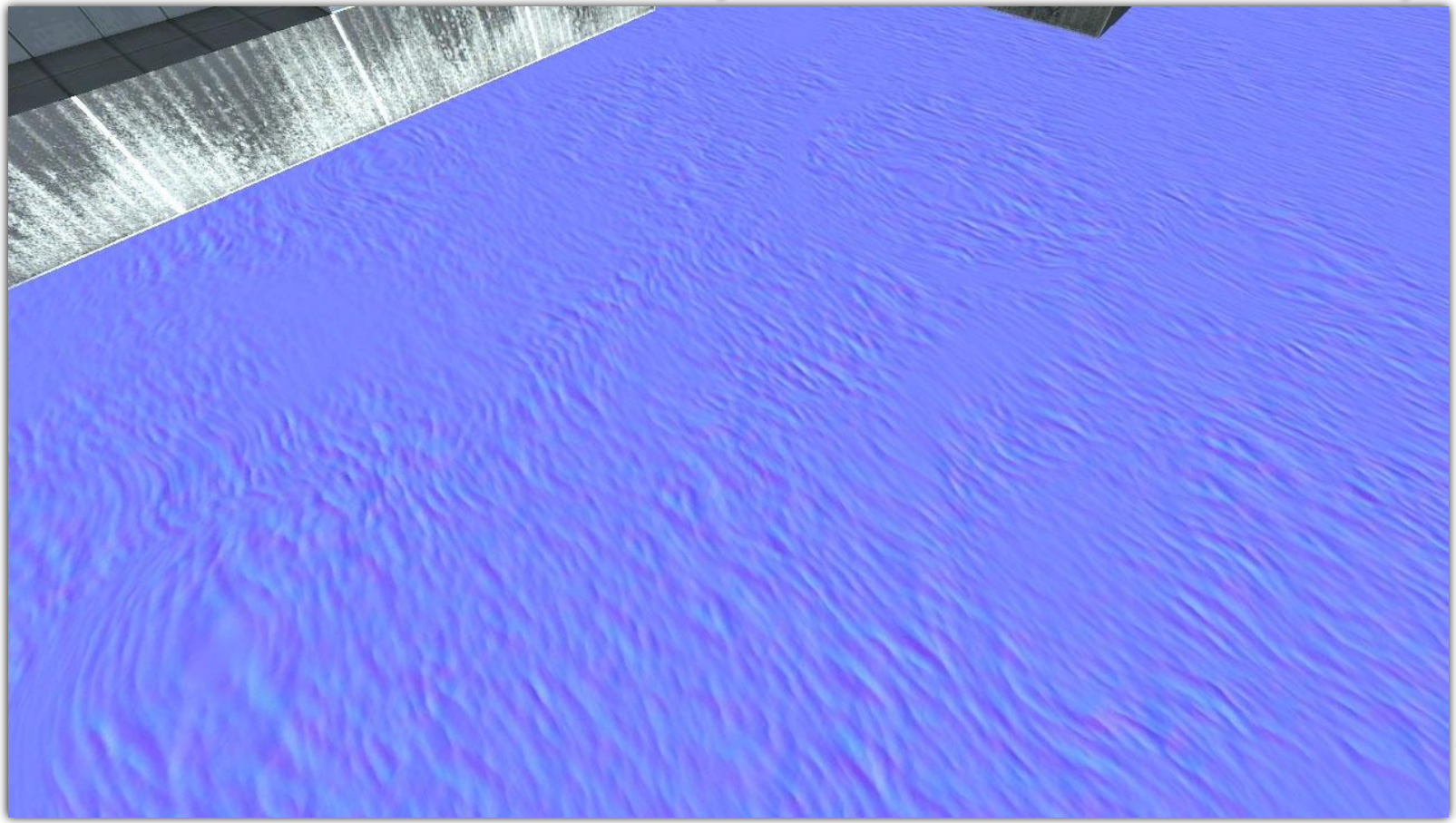
Debris Flow Example



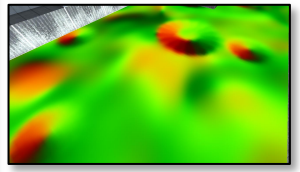
Flow Texture



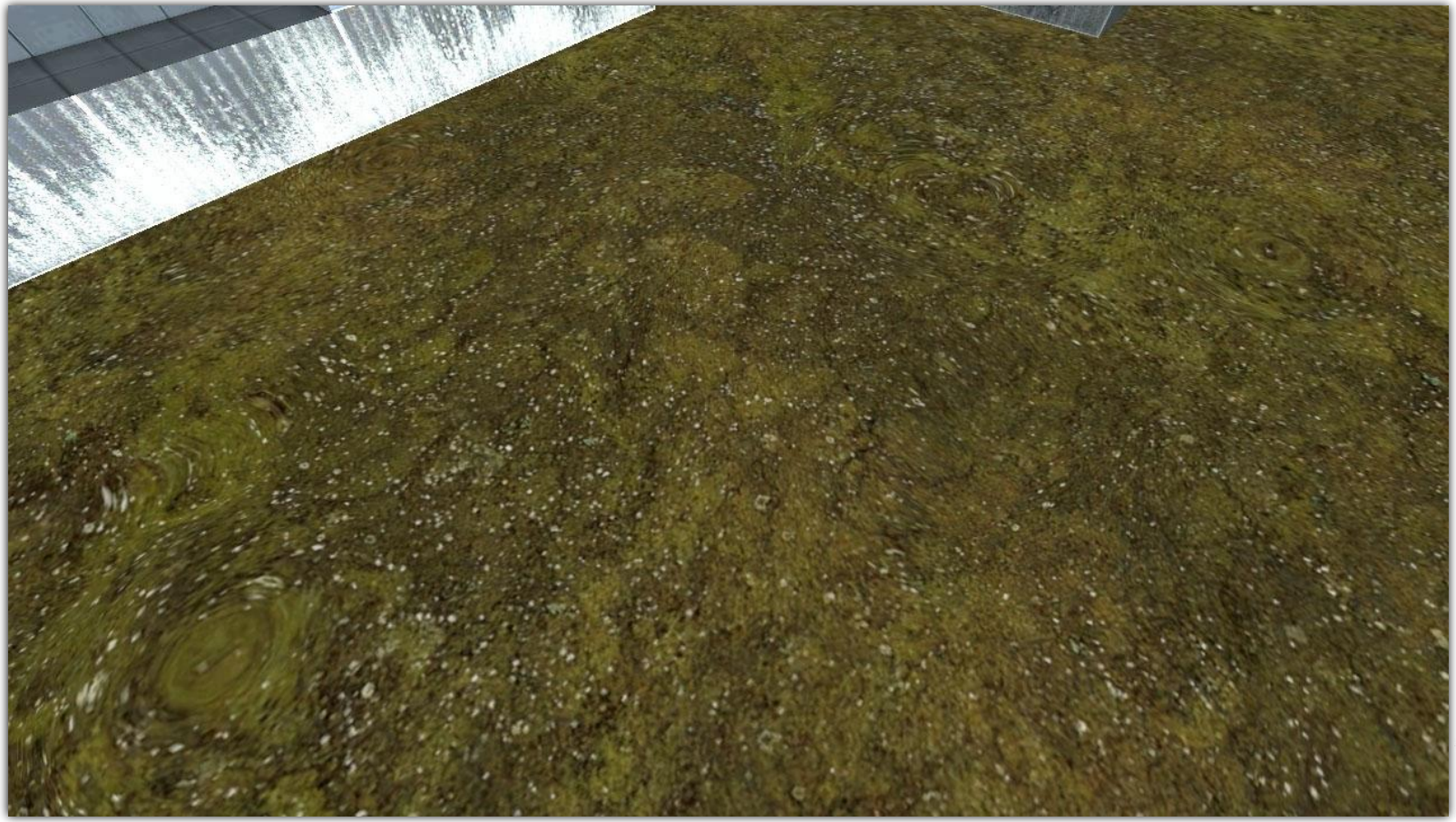
Debris Normal (Same as before)



Flow Vectors

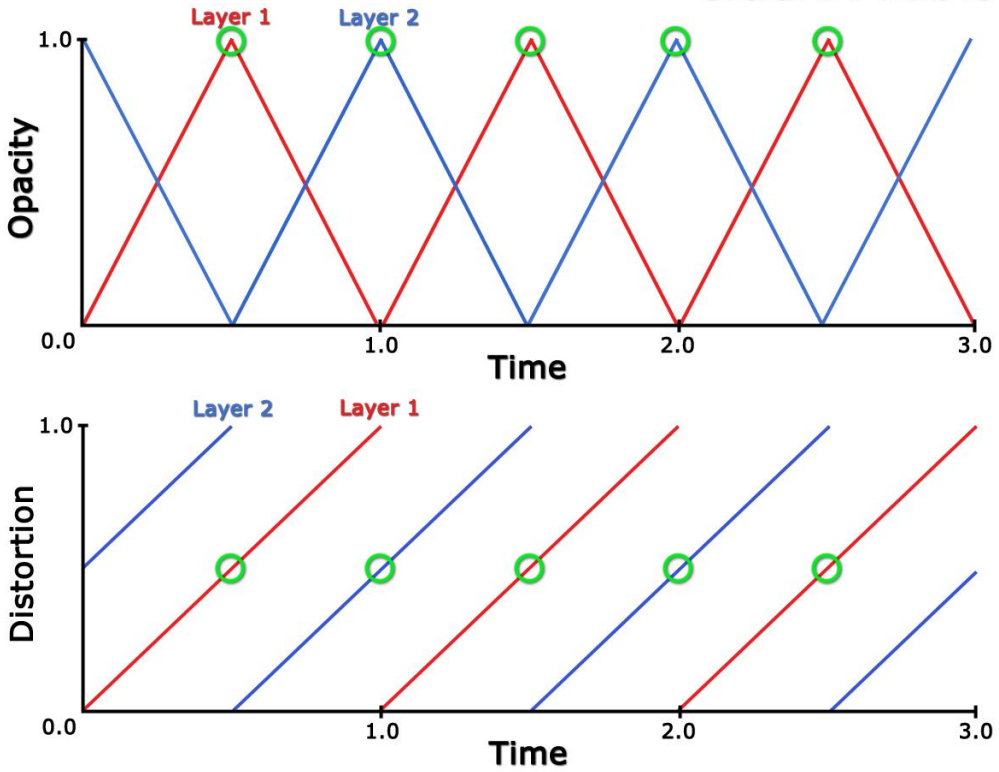


Flowing Debris Using Same Algorithm



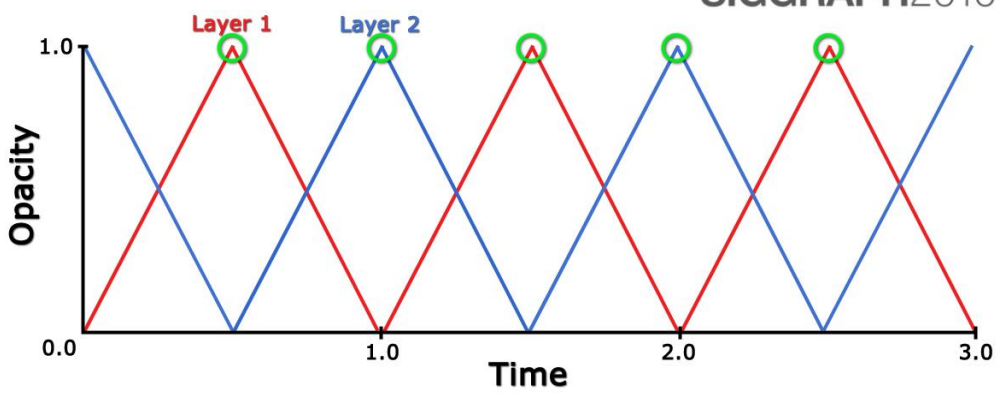
Flowing Normals

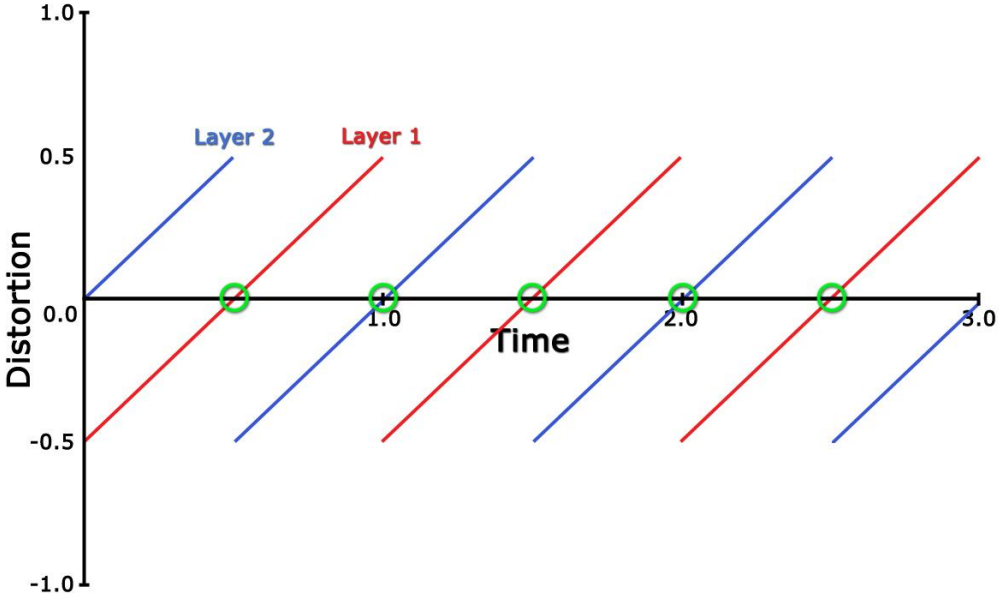
- Flowing normals would repeat an interval from zero to some fraction with the peak (center) of the interval at half distortion



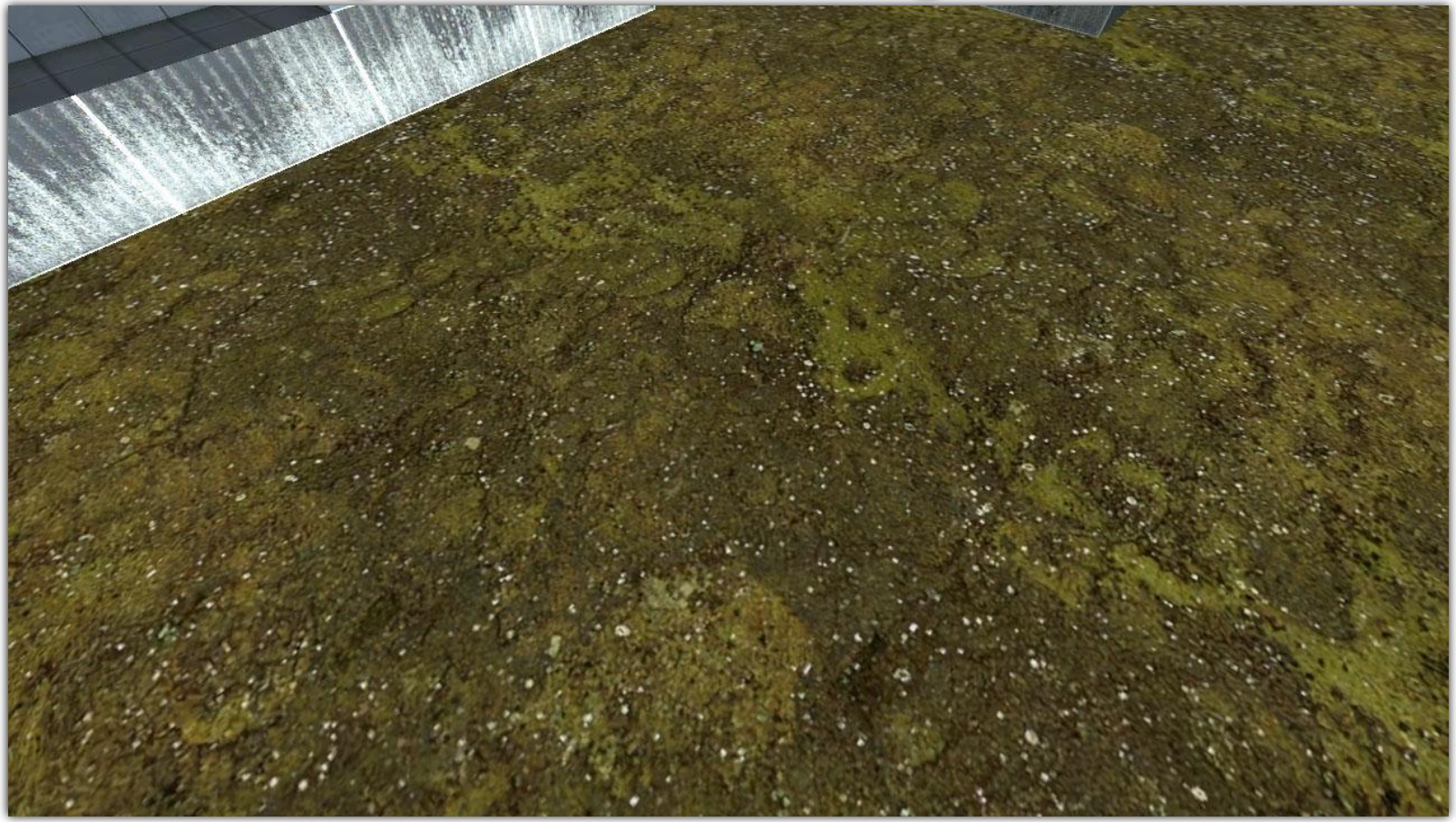
Flowing Debris

- Flowing colors works better by offsetting the interval from –`fraction` to `+fraction` so the peak of the interval is at zero (the at-rest position)

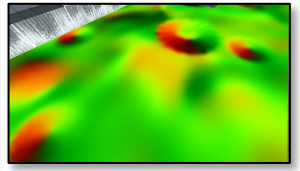




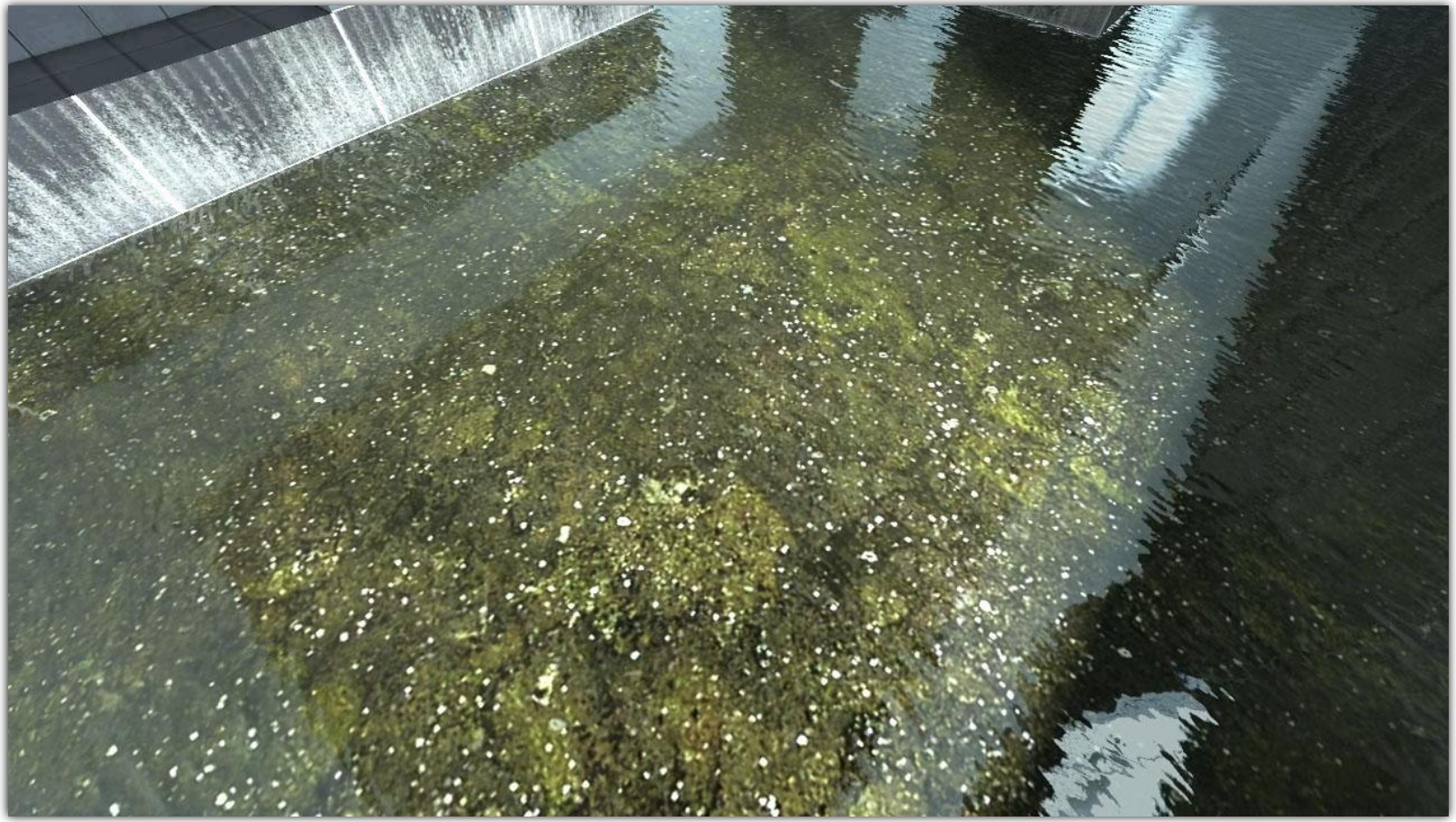
Flowing Debris Using Offset



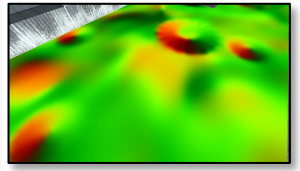
Flow Vectors



Debris Flow



Flow Vectors



Future Work

- Flow height maps and use tessellation hardware
- Multiple frequencies of normal maps
- Render dynamic flow vectors per-frame so animated objects cause flow changes
- Use flow map with our physics simulation to have objects flow on the water surface using the same data

Summary

- Use an artist-authored flow map
- Flow the normals in two layers and combine
- Use noise to reduce pulsing artifact
- Offset each phase of animation to reduce repetition
- Flowing debris uses an offset distortion range that favors less distortion than the normal flow

Thank You!

Water textures created by Alireza Razmpoosh

Alex Vlachos, Valve
alex@valvesoftware.com