River

This page provides a tutorial on creating a river simulation with Chaos Phoenix in Maya.

Overview

This is an Advanced Level tutorial. The workflow for setting up the shot, and the Phoenix settings involved in the simulation are explained in detail. However, creating a production quality shot of a similar nature may require some tweaks to the lighting, materials and/or the Phoenix simulation.

Requires Phoenix FD 3.12.00 Official Release and V-Ray Next Official Release for Maya 2015 or newer. You can download official Phoenix and V-Ray from https://download.chaos.com. If you notice a major difference between the results shown here and the behavior of your setup, please reach us using the Support Form.

The instructions on this page guide you through the process of using Phoenix to create a simulation of a river flowing down a hill.

The **Download** button below provides you with an archive containing the scene file.

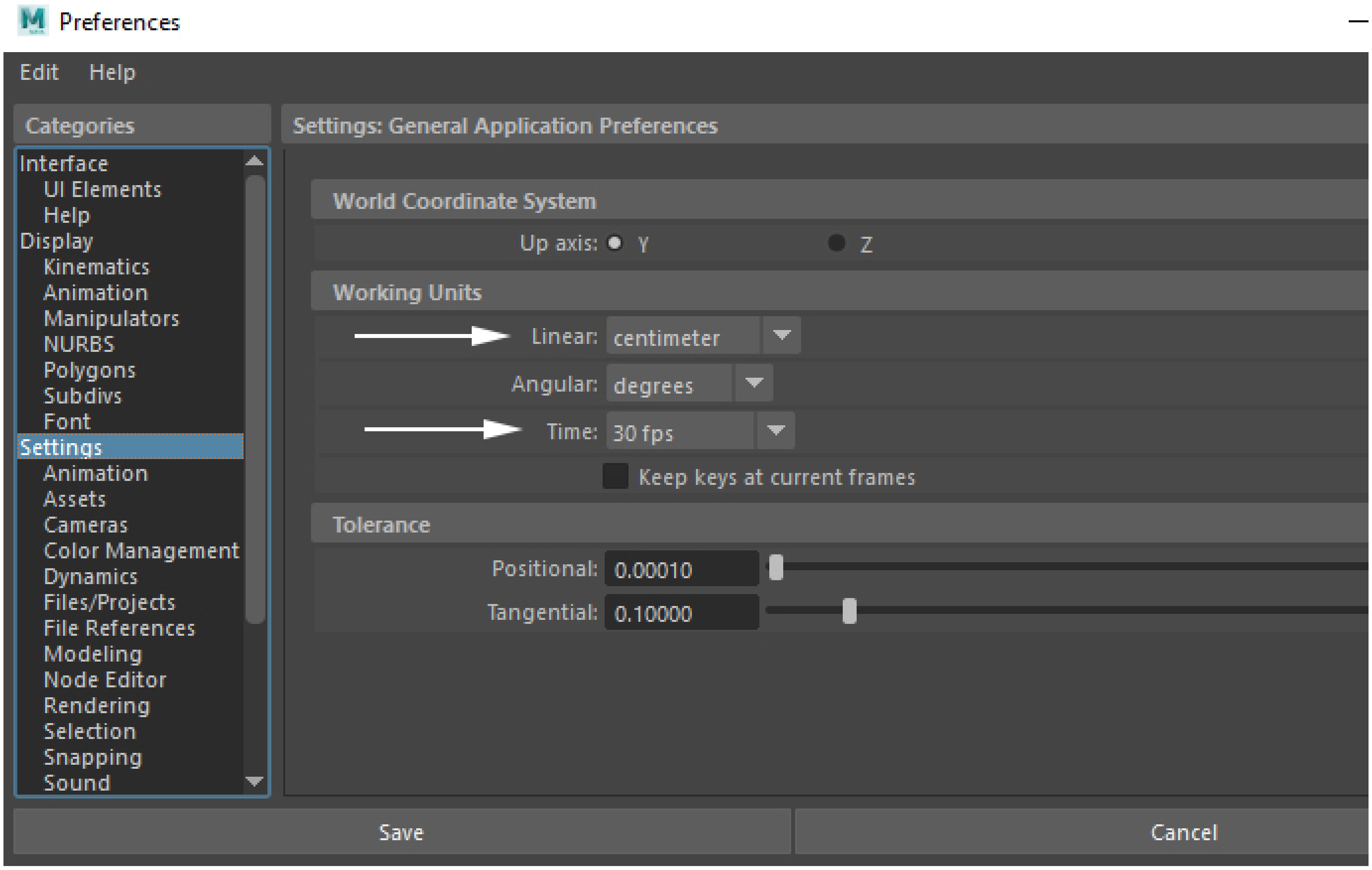
Download Project Files (84MB)

Units Setup

Scale is crucial for the behavior of any simulation. The real-world **size of the Simulator** in **units** is important for the simulation dynamics. Large-scale simulations appear to move more slowly, while mid-to-small scale simulations have lots of vigorous movement. When you create your Simulator, you must check the **Grid** rollout where the real-world extents of the Simulator are shown. If the size of the Simulator in the scene cannot be changed, you can cheat the solver into working as if the scale is larger or smaller by changing the **Scene Scale** option in the **Grid** rollout.

Go to **Windows Settings and Preferences Preferences Settings** and set the Working Units to **centimeters**.

Set the **Time** to **30fps**. We do this to ensure the setup is identical to the Burning Chair tutorial for 3ds Max. You may skip this step if you find it unnecessary. Keep in mind the final result of your simulation may turn out slightly different.



Settings: General Application Preferences

World Coordinate System

Up axis: ● Y ○ Z

Working Units

Linear: centimeter

Angular: degrees

Time: 30 fps

□ Keep keys at current frames

Tolerance

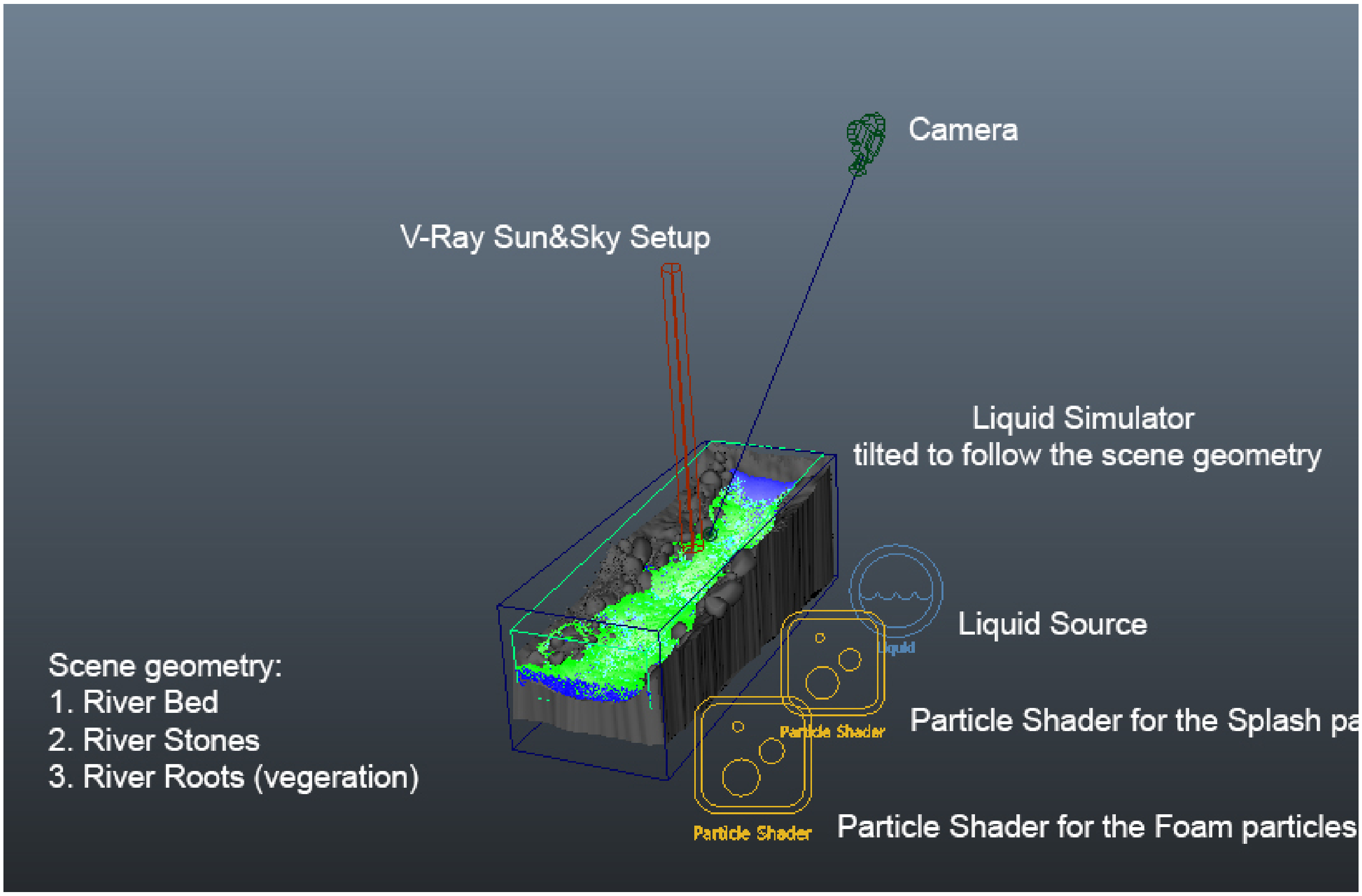
Positional: 0.00010

Tangential: 0.10000

Scene Layout

The final scene consists of the following elements:

1. A **Phoenix Simulator** tilted to follow the orientation of the scene geometry. **Initial Fill Up** under the **Liquid** rollout is enabled so waiting for the **Liquid Source** to fill the entire river bed geometry is not required.
2. **Scene geometry**: geometry for the **river bed**, **stones** and **vegetation**.
3. A **Phoenix Liquid Source** emitting from manually specified **faces** at the top of the **river bed** geometry.
4. Two **Phoenix Particle Shaders** used for the **Splash** and **Foam** particles. The **Splash** particles are rendered as **Points** while the **Foam** is rendered as **Bubbles**.
5. A **V-Ray Sun&Sky** setup for lighting the scene.
6. A **V-Ray Physical Camera** for rendering.

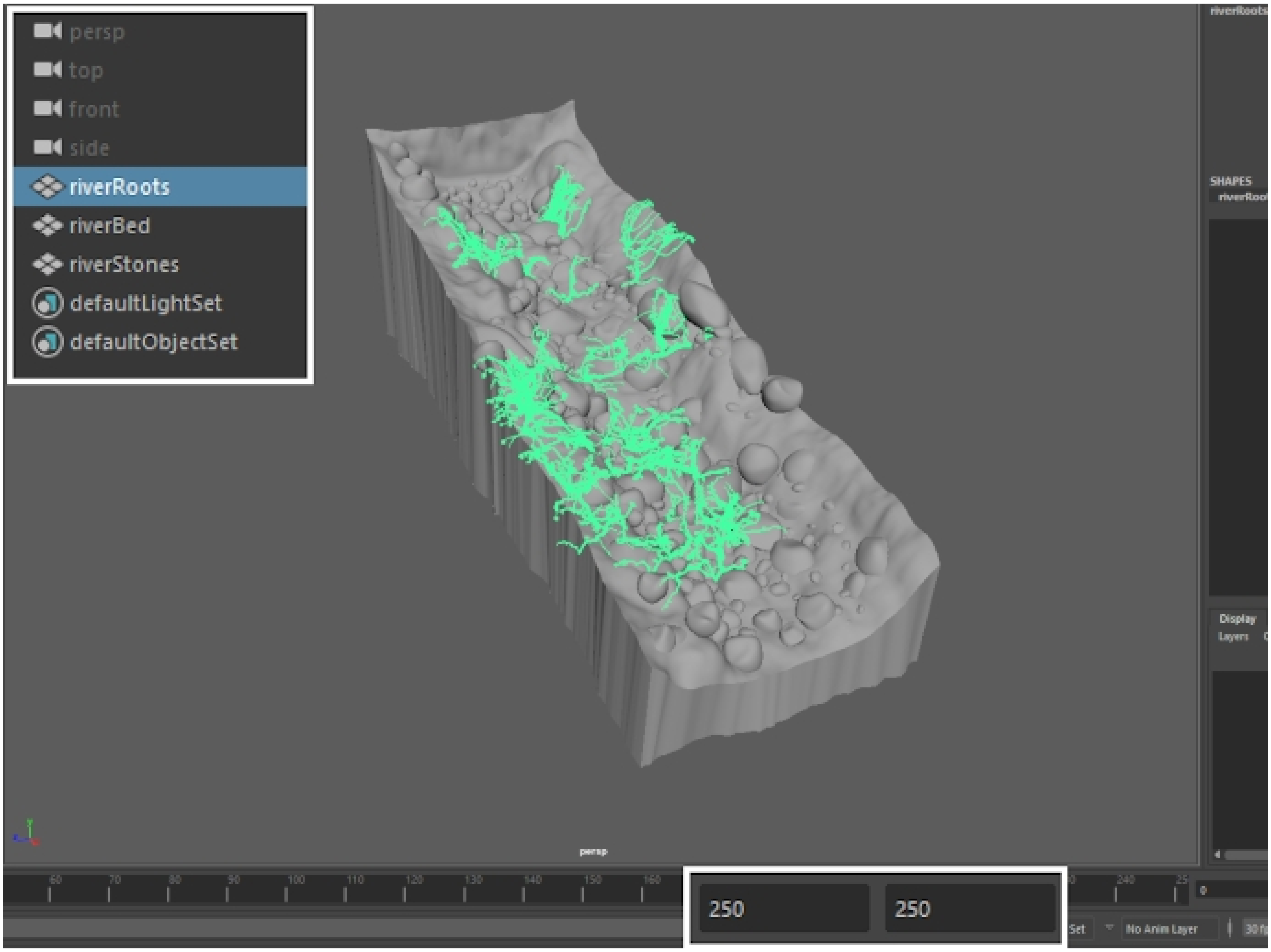

Camera
V-Ray Sun&Sky Setup
Liquid Simulator
tilted to follow the scene geometry
Liquid Source
Liquid
Particle Shader
Particle Shader for the Splash pa
Particle Shader
Particle Shader for the Foam particles
Scene geometry:
1. River Bed
2. River Stones
3. River Roots (vegeration)

Scene Setup

Set the **Time Slider** Frame Range to **0 - 250**.

Import the **riverBed.abc**, **riverRocks.abc** and **riverRoots.abc** files to your scene by going to **Cache Alembic Cache Import Alembic...**

We use the **riverBed** and **riverRocks** as **collision geometry** for the Phoenix Liquid simulation while the riverRoots are only used during rendering.



Simulation Setup

Create a **Phoenix Simulator** and set the **Grid Cell Size** to **1. 953cm**.

Set the **Simulation Template** to **Liquid** from the drop-down at the top of the Phoenix Simulator parameters in the Attribute Editor.

By default, the Phoenix Simulator is set up for Fire/Smoke simulations. When the Liquid preset is selected, the FLIP solver is enabled instead (to manually do this, go to Liquid Enabled) and the Rendering Render Mode is set to Mesh.

Set the **Scene Scale** to **1**.

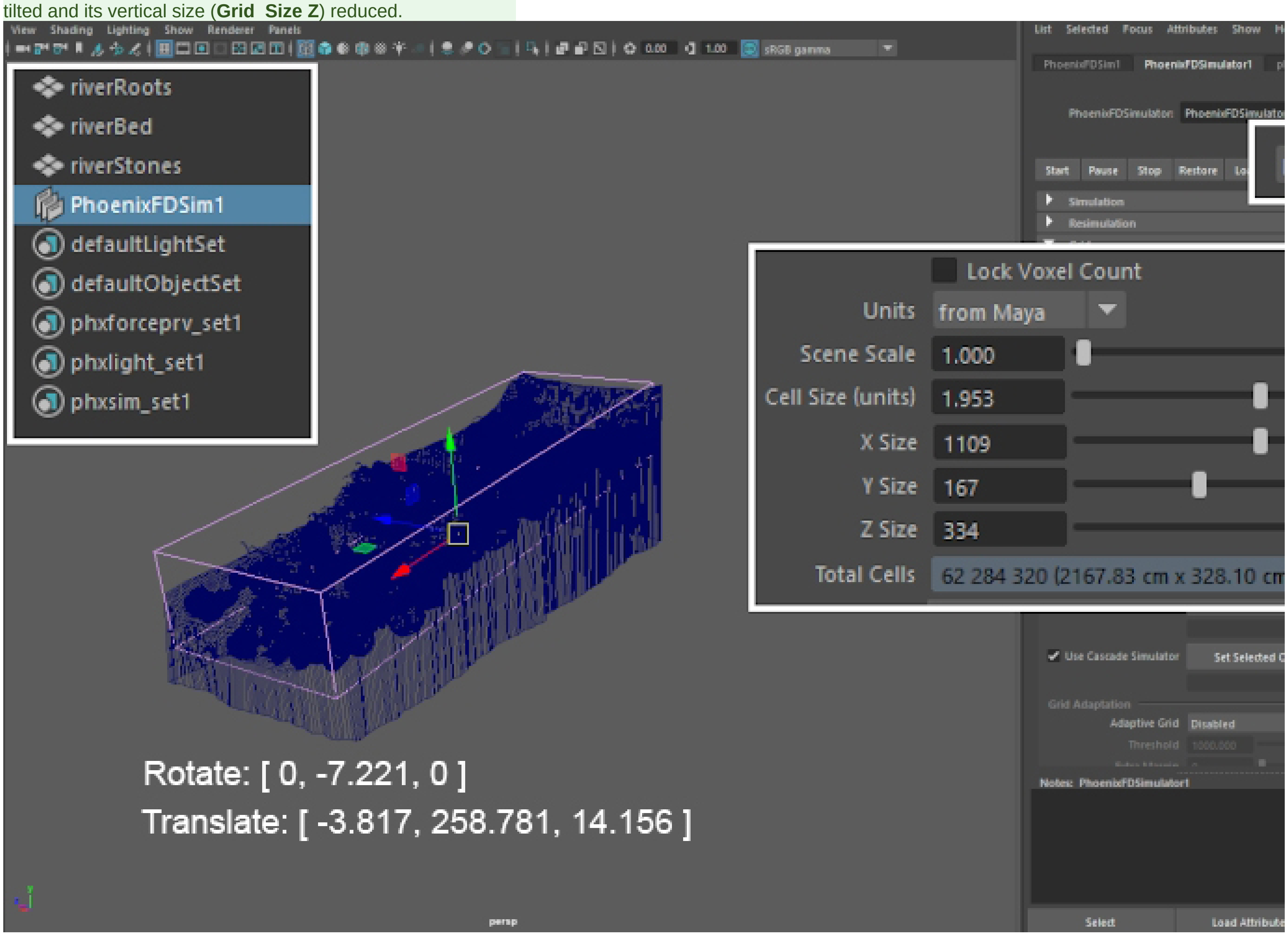
Set the **Grid Size X/Y/Z** to **1109 / 334 / 167** respectively.

For this example, the Simulator is placed at the following coordinates:

> **Translate X/Y/Z:** [-3.817, 258.781, 14.156]
> **Rotate X/Y/Z:** [0, 0, -7.221]

The simulation bounding box should now be wrapped around the **riverBed** geometry and slightly **tilted** to follow the direction of the stream.

We do this for the slight performance boost - keeping the Simulator straight provides no benefit - the simulation will be exactly the same but run faster when the bounding box is tilted and its vertical size (**Grid Size Z**) reduced.



□ Lock Voxel Count

Units: from Maya

Scene Scale: 1.000

Cell Size (units): 1.953

X Size: 1109

Y Size: 167

Z Size: 334

Total Cells: 62 284 320 (2167.83 cm x 328.10 cm

☑ Use Cascade Simulator

Adaptive Grid: Disabled

Threshold: 1000.000

Rotate: [0, -7.221, 0]

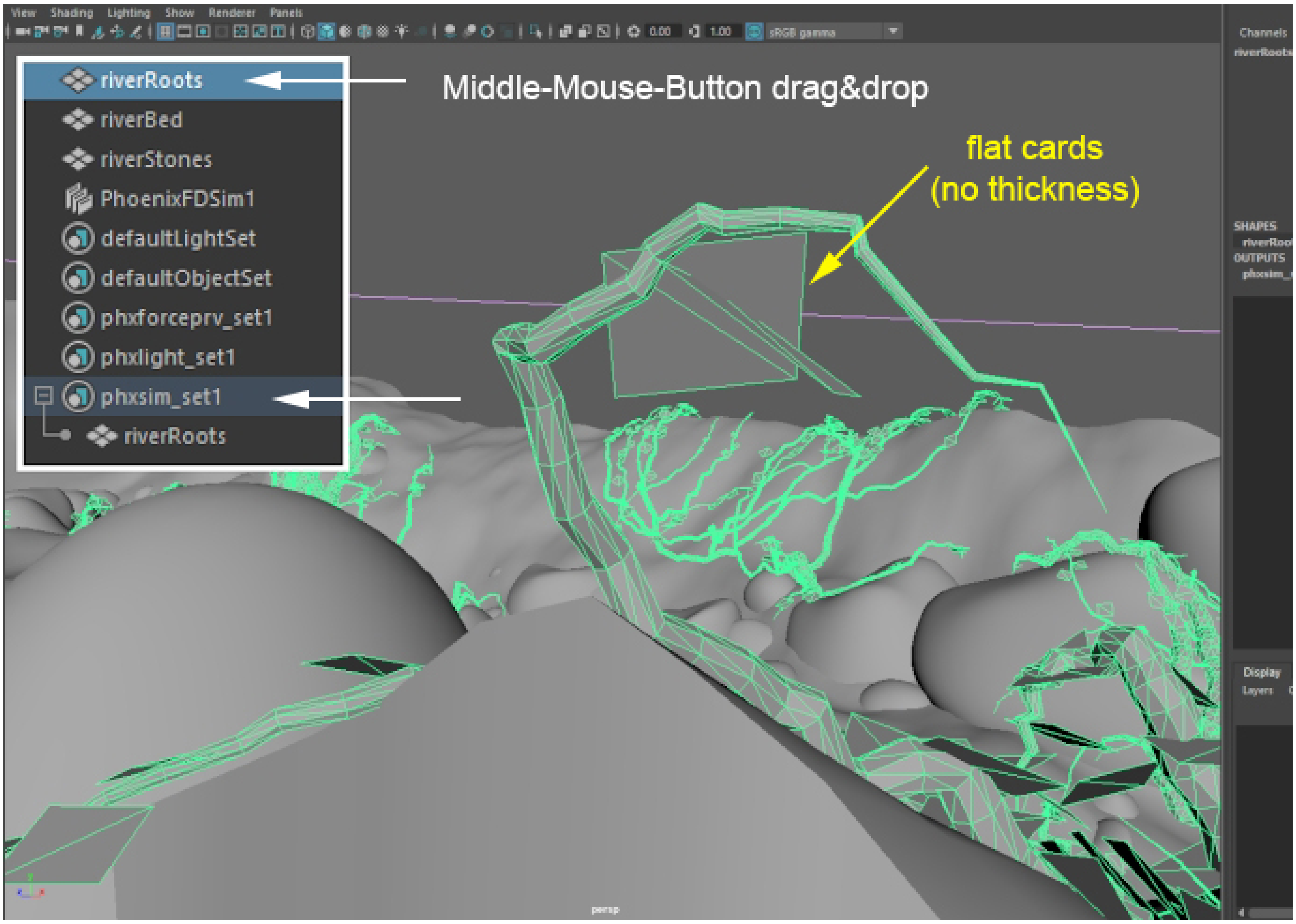
Translate: [-3.817, 258.781, 14.156]

Add the **riverRoots** geometry to **phxsim_set1** by Middle-Mouse-Button drag&dropping it in the **Outliner**.

The automatically generated **simulation set** can be used to specify which objects in the scene will **interact** with the **Phoenix Simulation**.

By default, the set acts as an **Exclude List** (**Simulator Scene Interaction** rollout), meaning that objects inside the set will be ignored during the simulation.

Using open geometry or geometry with no thickness (such as the poly planes used for the riverRoots' leaves in this example) can give you unpredictable simulation results. Making sure that your geometry is clean is crucial for a smooth workflow. Phoenix (and many simulation packages in general) use a volumetric representation of your mesh for the simulation. The process of creating this volumetric representation is called *voxelization*. The algorithms responsible for *voxelizing* the geometry can fail when using open (with holes) or planar (no thickness) geometry.

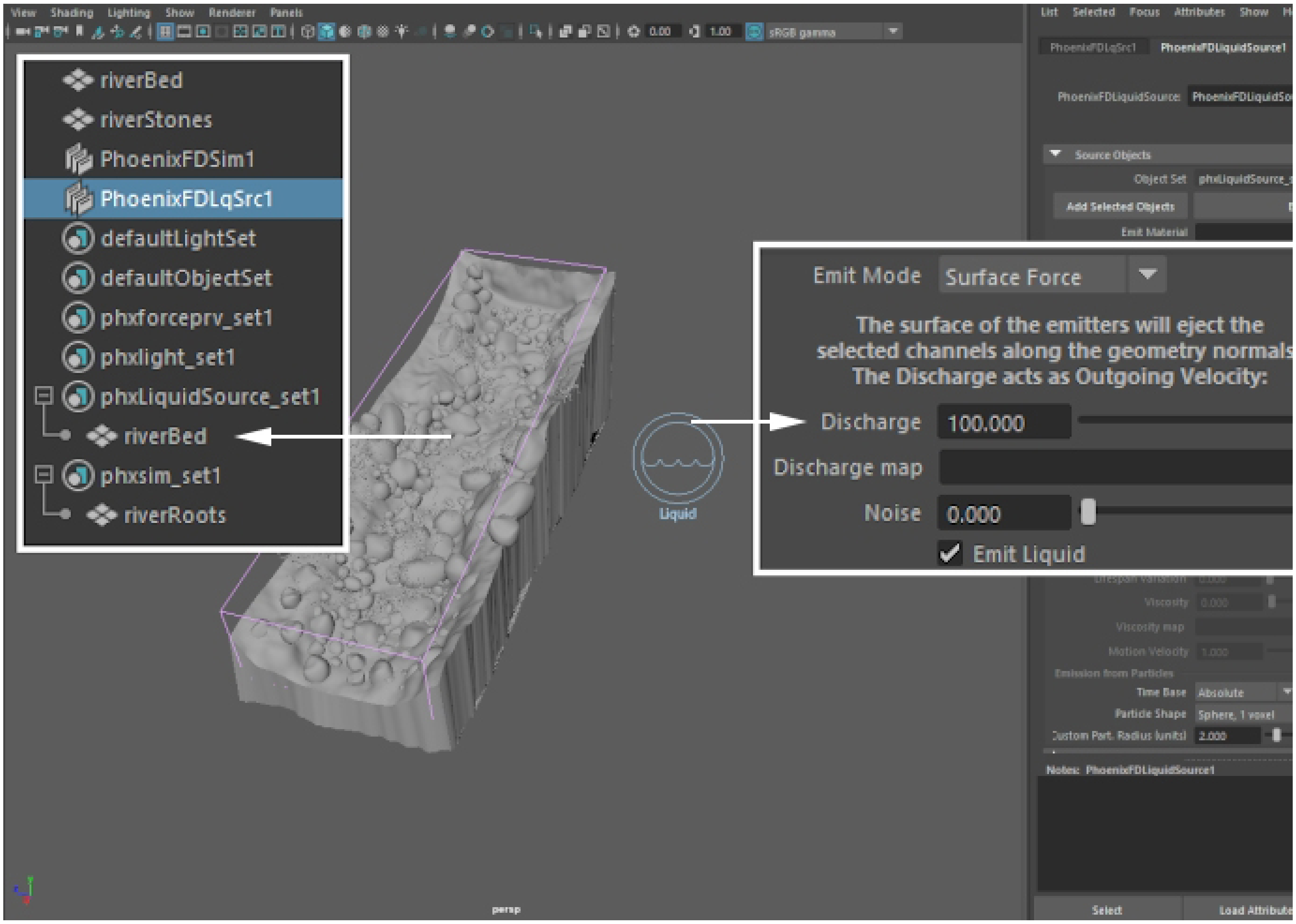


Middle-Mouse-Button drag&drop

flat cards
(no thickness)

Create a **Phoenix Liquid Source** and **add** the **riverBed** geometry to the Liquid Source's **set**. This automatically generated set is used to **specify the objects used for emission** by the Source.

Set the **Outgoing Velocity** to **100cm**.



Emit Mode Surface Force

The surface of the emitters will eject the selected channels along the geometry normals The Discharge acts as Outgoing Velocity:

Discharge 100.000

Discharge map

Noise 0.000

☑ Emit Liquid

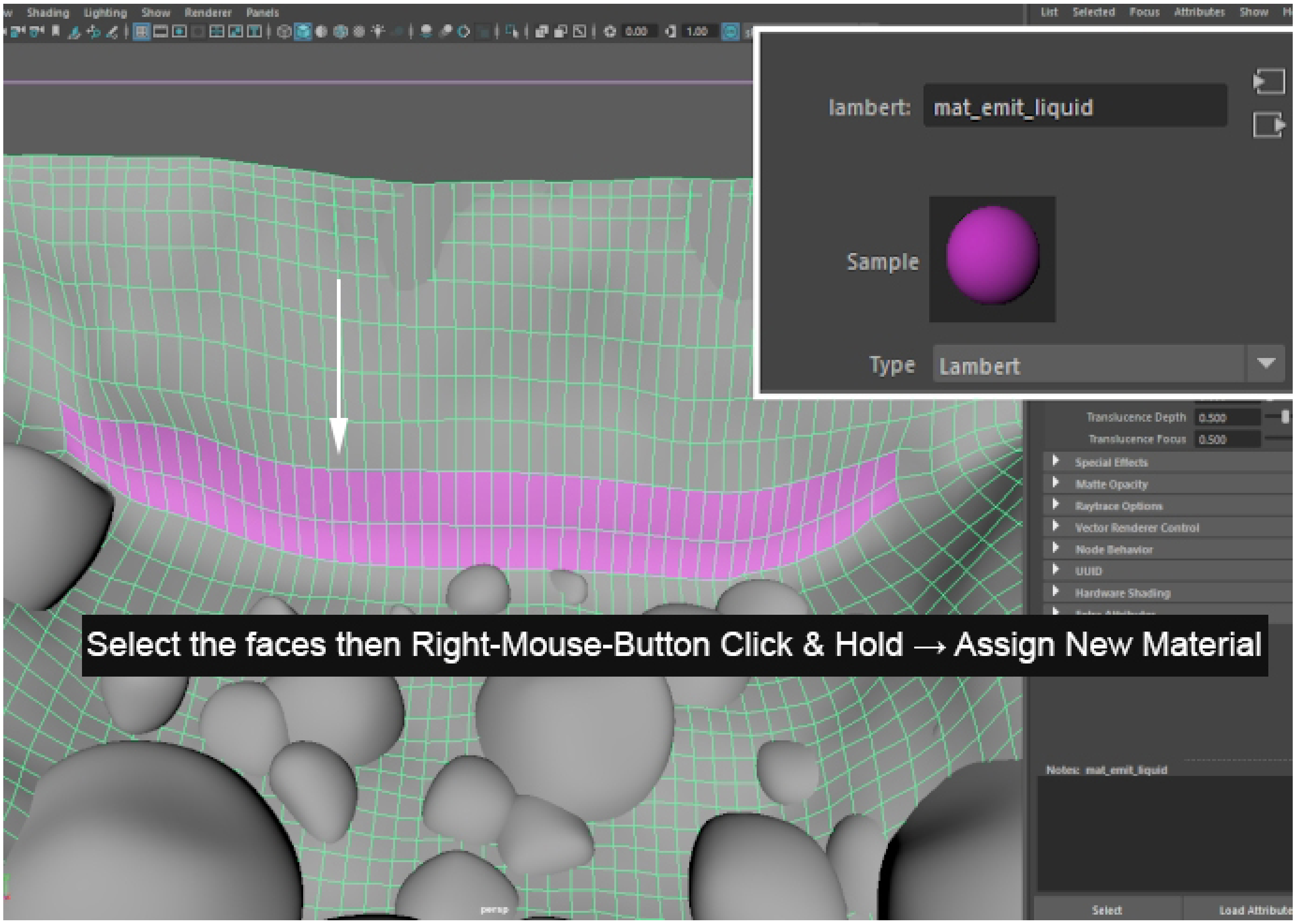
Here's a preview of the simulation in its current state.

The **entire riverBed** geometry is **emitting liquid** into the Simulator. Instead, we want liquid to be emitted into the simulation only at the **very start** of the river.

We can accomplish this by giving the **Phoenix Liquid Source** a **set of faces** specified by us to use for the emission. The **E mit Material** parameter at the top of the options panel for the **Source** can be used for this.

Select the **riverBed** geometry and go into **Face Selection** mode (F11). Select the faces at the very **start of the stream** and **assign** a **new Lambert material** to them.

Rename the new material to **mat_emit_liquid** and give it a fabulous **Diffuse Color**.



Select the faces then Right-Mouse-Button Click & Hold → Assign New Material

lambert: mat_emit_liquid

Type: Lambert

Translucence Depth: 0.500

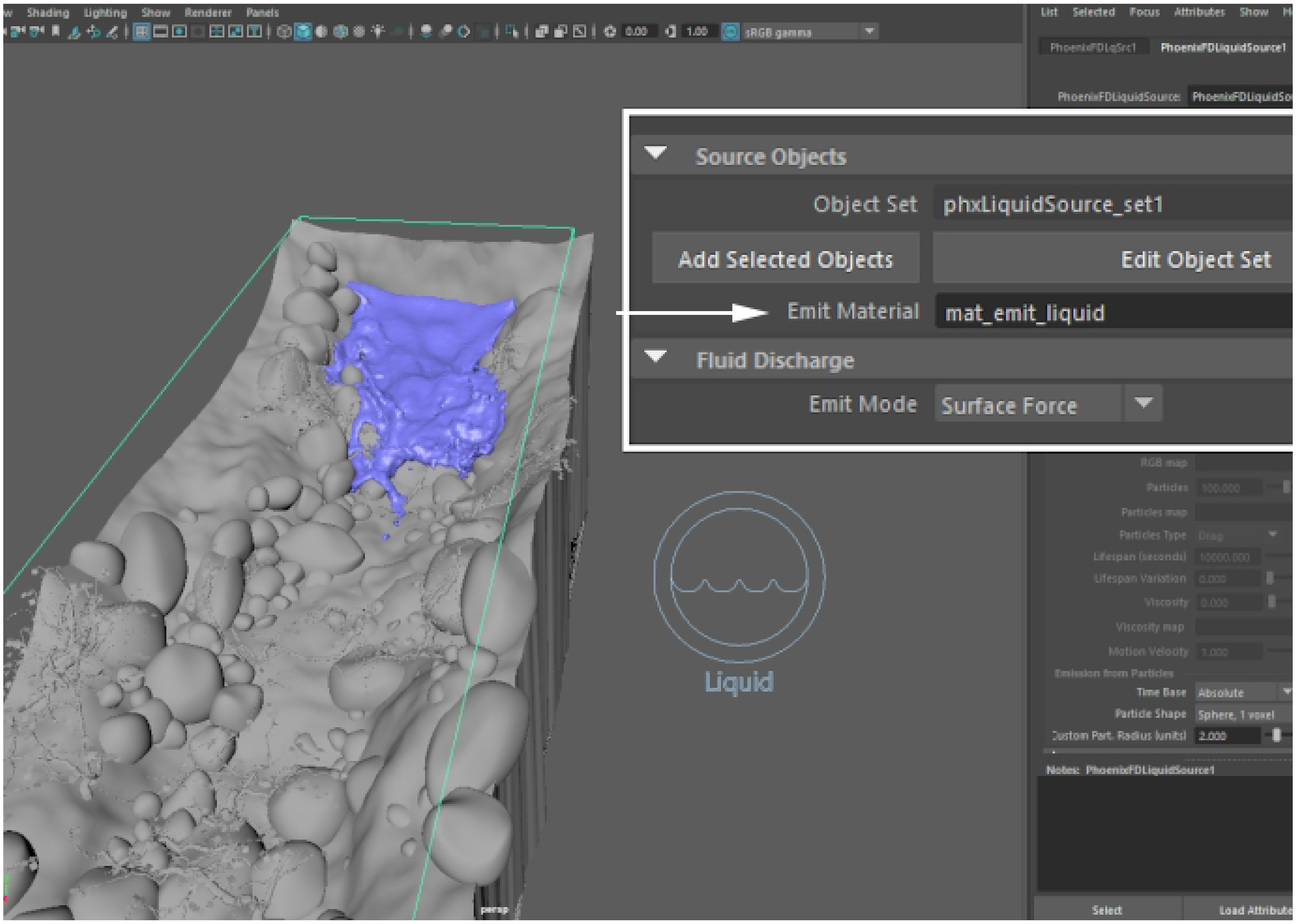
Translucence Focus: 0.500

Notes: mat_emit_liquid

Select the **Phoenix Liquid Source** and set the **Emit Material** parameter to point to **mat_emit_liquid**. You can do this either by:

1. Manually typing the name in the Emit Material field and hitting enter.
2. Middle-Mouse-Button drag&dropping the material from the Hypershade window.

The **Source** will now only emit from **faces** that share this **material**.



Source Objects

Object Set phxLiquidSource_set1

Emit Material mat_emit_liquid

Fluid Discharge

Emit Mode Surface Force

RGB map

Particles 100.000

Particles map

Particles Type Drag

Lifespan (seconds) 10000.000

Lifespan Variation 0.000

Viscosity 0.000

Viscosity map

Motion Velocity 1.000

Emission from Particles

Time Base Absolute

Particle Shape Sphere, 1 voxel

Custom Part. Radius (units) 2.000

Notes: PhoenixFDLiquidSource1

Here's a preview of the simulation in its current state.

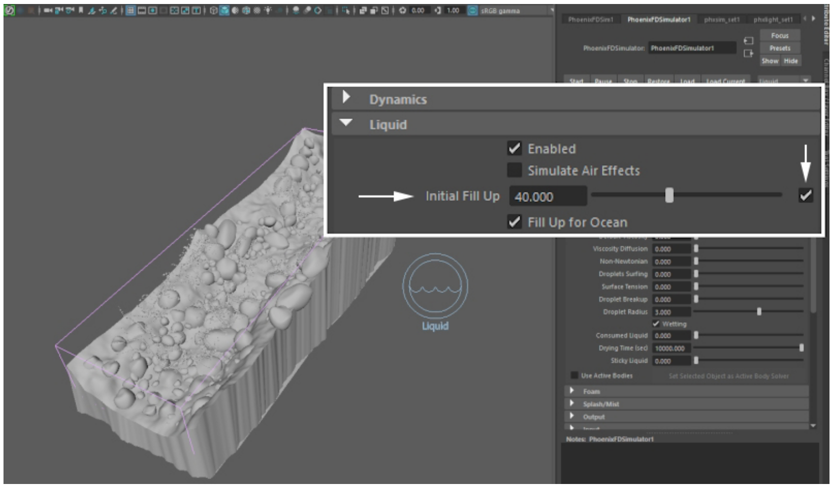
One major issue at the moment is the fact that it takes **about 250 frames** for the river bed to fill up with water. This can become a major issue when running high-res simulations and tweaking individual parameters for the Foam, Splashes, etc.

There are two ways to address this:

1. Run the simulation for 200 or so frames and use the **Simulation** rollout **Load & Start** button to start a new simulation from **the last frame of the one before**.
2. Enable **Dynamics Initial Fill Up** to fill the Liquid Simulator with water on the first frame of the simulation.

We go with the second approach.

Enable Liquid **Initial Fill Up** and set the level to **40** (% of the simulator bounding box height).



▸ Dynamics

▾ Liquid

☑ Enabled

☐ Simulate Air Effects

Initial Fill Up 40.000 ☑

☑ Fill Up for Ocean

Viscosity Diffusion 0.000

Non-Newtonian 0.000

Droplets Surfing 0.000

Surface Tension 0.000

Droplet Breakup 0.000

Droplet Radius 3.000

☑ Wetting

Consumed Liquid 0.000

Drying Time (sec) 10000.000

Sticky Liquid 0.000

☐ Use Active Bodies

Liquid
PhoenixFDSimulator1
Focus
Presets
Show
Hide
Set Selected Object as Active Body Solver
Foam
Splash/Mist
Output
Notes: PhoenixFDSimulator1

Here's a preview of the simulation in its current state.

Before we take care of the **Foam / Splash** particles, we take a quick look at **Wetting** in the next step.

Wetting

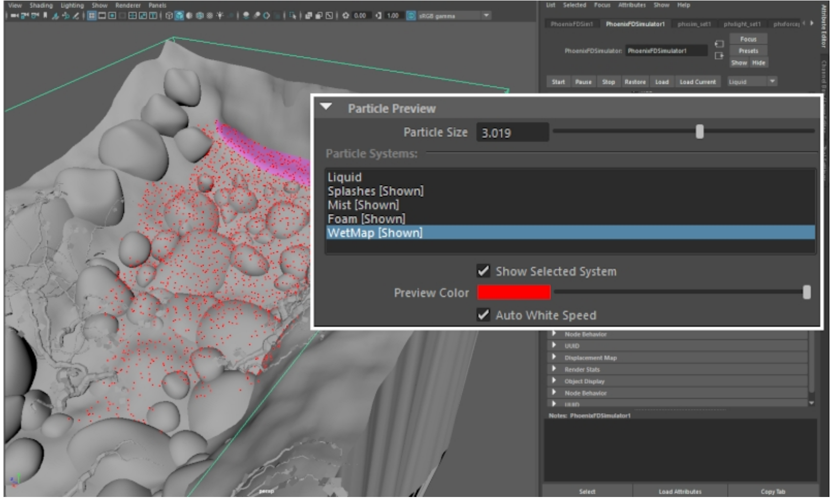
For a realistic interaction between the scene geometry and the liquid, Phoenix provides you with the option to simulate the effect of **Wetting**.

Wetting is handled by a set of secondary **WetMap Particles** which are calculated along with the **Liquid**, **Foam**, **Splashes**, etc.

WetMap particles are created where the **Liquid** particles **interact** with the scene **geometry**.

You can use the **WetMap Particles** through a **Phoenix Particle Texture** to generate a **Black&White Texture Map** which can be used as a mask for a **Blend Material**.

The **preview** of **WetMap Particles** can be Enabled/Disabled from the Phoenix Simulator **Preview rollout Particle Preview** options.



▾ Particle Preview

Particle Size 3.019

Particle Systems:

Liquid
Splashes [Shown]
Mist [Shown]
Foam [Shown]
WetMap [Shown]

☑ Show Selected System

Preview Color 🟥

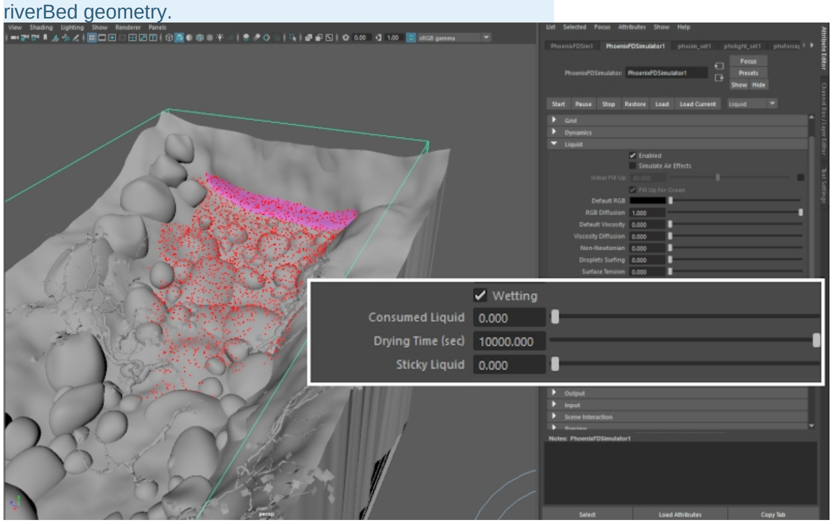
☑ Auto White Speed

To **enable** the simulation of **WetMap Particles**, **enable Wetting** from the **Simulator Liquid** rollout.

Wetting is a simulation, not a render-time option - changing any of the values below requires a re-run of the simulation.

Run the simulation for 30 or so frames.

Phoenix Simulator Liquid **Initial Fill Up** is **Disabled** for clarity. The only source of liquid is the polygons on the riverBed geometry.



PhoenixFDSimulator: PhoenixFDSimulator1

Liquid

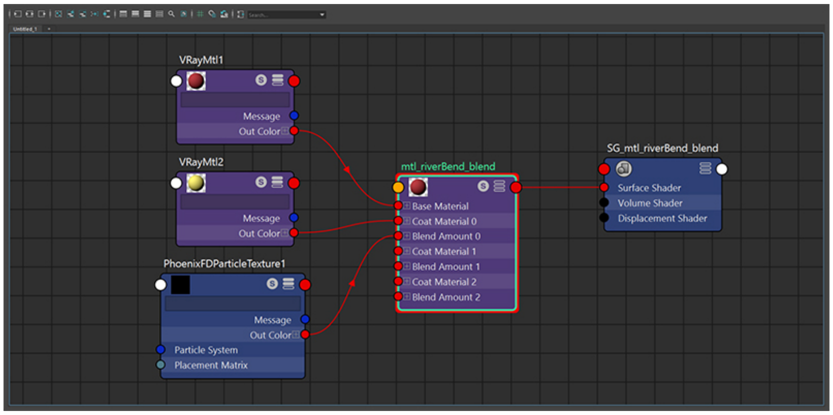
☑ Enabled
□ Simulate Air Effects
Initial Fill Up 40.000
☑ Fill Up for Ocean
Default RGB
RGB Diffusion 1.000
Default Viscosity 0.000
Viscosity Diffusion 0.000
Non-Newtonian 0.000
Droplets Surfing 0.000
Surface Tension 0.000

☑ Wetting
Consumed Liquid 0.000
Drying Time (sec) 10000.000
Sticky Liquid 0.000

Assign a **V-Ray Blend Material** to the **riverBed** geometry.

Plug **two V-Ray materials** to the **Base** and **Coat 1** inputs of the **Blend** material. Set the **Diffuse** colors of those V-Ray materials to **Red** and **Yellow** respectively.

Plug a **Phoenix Particle Texture** to the **Blend 1** input of the **Blend** material - the **Particle Texture** will specify where the **C oat** material will override the **Base** one.

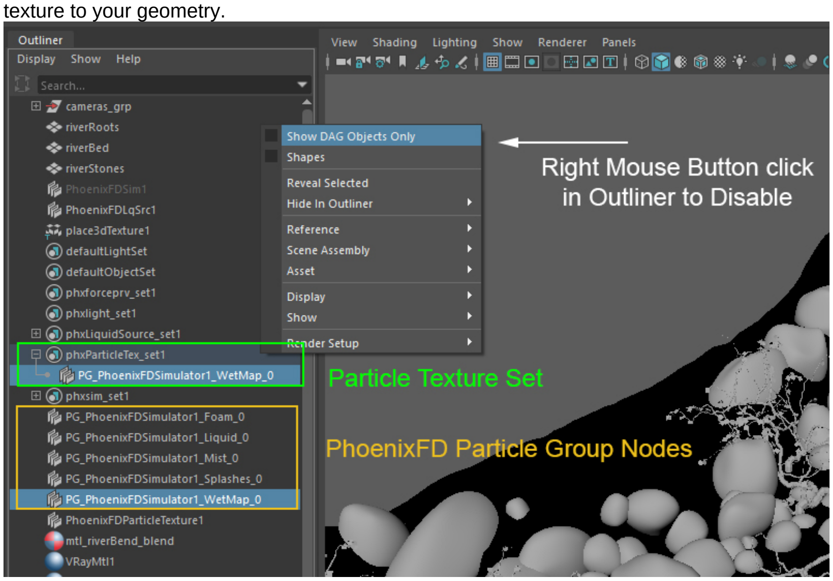


Right Mouse Button click on an **empty space** in the **Outliner** and **Disable** the **Show DAG Objects Only** option. The Phoenix Particle Groups generated by the Simulator are not visible by default - disabling this option will allow you to select them.

When created, the **Phoenix Particle Texture** generates a **set** used to **specify** which **particle groups** will be evaluated. You need to provide the WetMap Particle Group to the set so the Particle Texture can use it.

Middle Mouse Button click&drag the **PG_PhoenixFDSimulator1_WetMap_0** into the set **phxParticleTex_set1**.

A **3d Placement Node** is also generated automatically. Do not delete it - Phoenix needs this node to properly map the texture to your geometry.

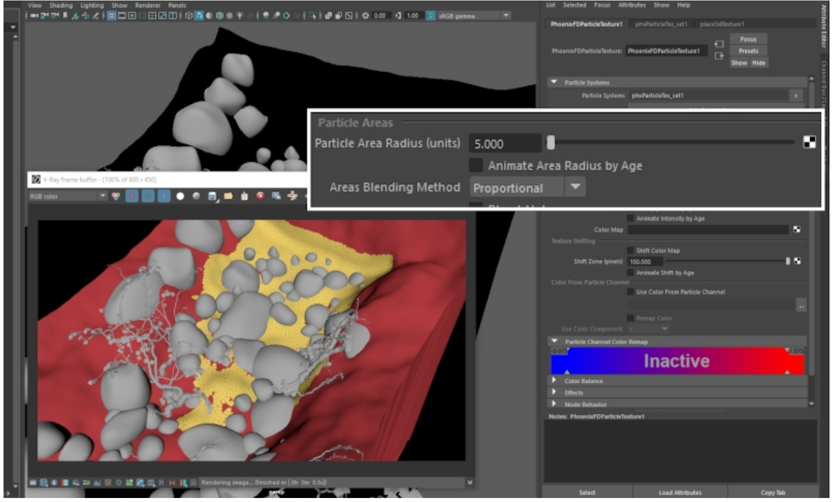

Right Mouse Button click in Outliner to Disable

Particle Texture Set

PhoenixFD Particle Group Nodes

Increase the **Particle Area Radius (units)** parameter until the dots generated by each WetMap particle start blending together.

For this example, the value is set to **5**.



Splash and Foam

We are now ready to start setting up the **Splash** and **Foam** parameters for the **Phoenix Simulator**.

Open the **Splash/Mist** rollout and select the **Enable** checkbox.

Choose **Yes** when asked if you'd like a **Phoenix Particle Shader** generated for the **Splash** particles. This will automatically set up the link between the splash particle group, the Particle Shader, and the Liquid Simulator.

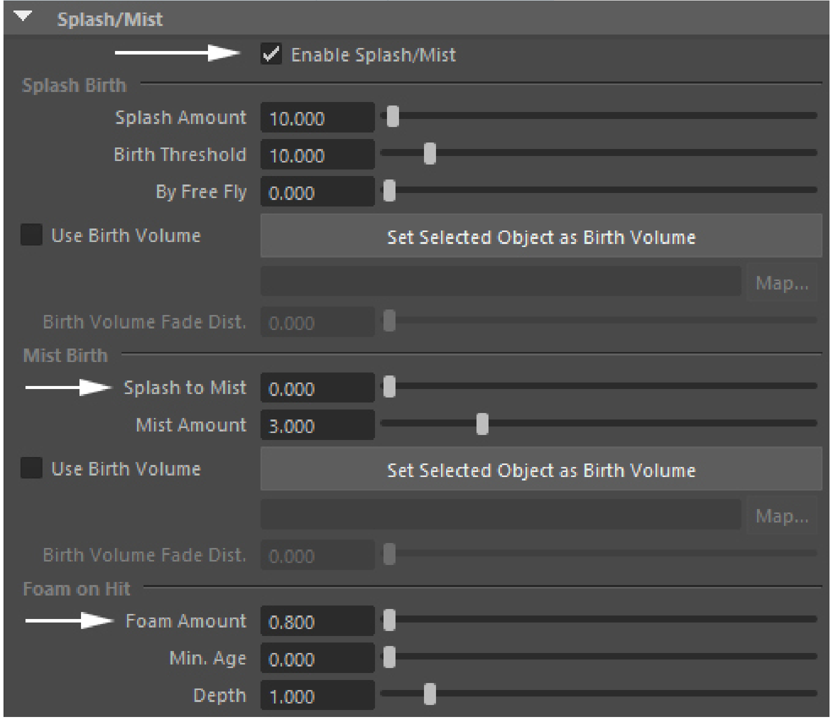
Set the **Splash to Mist** option **0** to completely disable the generation of Mist. Mist particles are produced by the splitting the Splash particles as they fly through the air.

Set the **Foam on Hit Foam Amount** option to **0.8**.

In this tutorial, the **Foam** is **generated by** the **Splash particles**. Splash particles, on the other hand, are born when the **liquid surface is turbulent**. Therefore, when the liquid collides with the river stones and starts twisting around them, Splash particles will be born which will produce Foam for you.

We don't use directly give birth to Foam particles using the Foam rollout Foam Amount option because it relies on the **Speed** of the liquid. Since the river is flowing down at a relatively constant rate, this will produce Foam all over the place rather than in those specific areas where the liquid flow is disturbed.

If you'd like to learn more about the different secondary particle effects and how they interact with the simulation, you can check the FLIP Particles Life Cycle diagram.



Splash/Mist

☑ Enable Splash/Mist

Splash Birth

Splash Amount 10.000

Birth Threshold 10.000

By Free Fly 0.000

□ Use Birth Volume — Set Selected Object as Birth Volume

Map...

Birth Volume Fade Dist. 0.000

Mist Birth

Splash to Mist 0.000

Mist Amount 3.000

□ Use Birth Volume — Set Selected Object as Birth Volume

Map...

Birth Volume Fade Dist. 0.000

Foam on Hit

Foam Amount 0.800

Min. Age 0.000

Depth 1.000

Here's a preview of the Splash particles on their own:

In the next step, we enable the simulation of Foam.

Open the **Liquid Simulator   Foam** rollout and select the Enable checkbox.

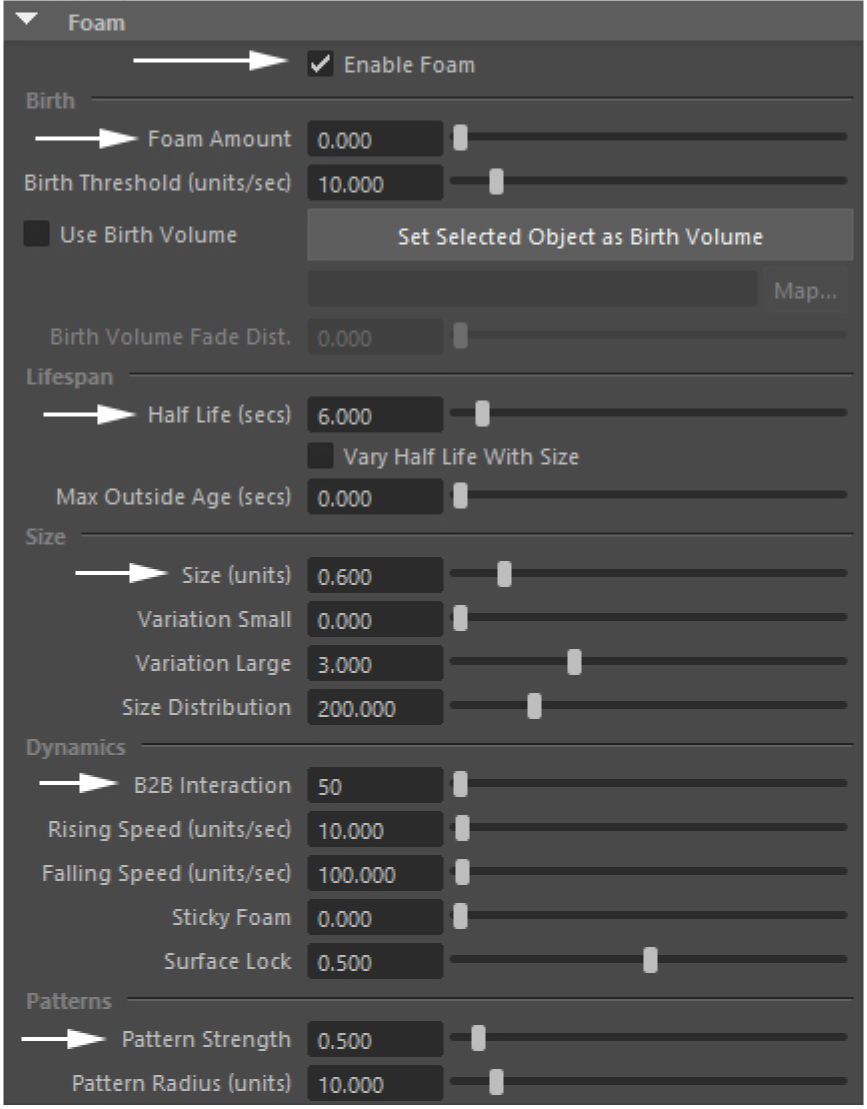
Set the **Foam Amount** to **0**. As discussed earlier, Foam particles should only be born by the **Splash** particles in this setup.

Set the **Lifespan Half Life** to **6** seconds. After some trial and error, we decided on this value. This may be changed for your scene if you are following along with a different setup.

Set the **Size** to **0.6cm**. The size of the liquid particles will affect how they interact with each-other when the **B2B Interaction** (Bubble-To-Bubble) is enabled (i.e. the value is greater than 0). Smaller Size will produce smaller clusters of particles and vice-versa.

Set the **B2B Interaction** to **50**. The Bubble-To-Bubble interaction helps the Foam particles **stick together** rather than behaving like individual points in space.

Set the **Patterns Strength** to **0.5** and the **Patterns Radius** to **10cm**. Those two options are entirely an artistic choice - please check the Example Videos in the Liquid Simulator's Foam rollout documentation for a more intuitive understanding of those controls.



Foam

☑ Enable Foam

Birth

Foam Amount 0.000

Birth Threshold (units/sec) 10.000

□ Use Birth Volume

Birth Volume Fade Dist. 0.000

Lifespan

Half Life (secs) 6.000

□ Vary Half Life With Size

Max Outside Age (secs) 0.000

Size

Size (units) 0.600

Variation Small 0.000

Variation Large 3.000

Size Distribution 200.000

Dynamics

B2B Interaction 50

Rising Speed (units/sec) 10.000

Falling Speed (units/sec) 100.000

Sticky Foam 0.000

Surface Lock 0.500

Patterns

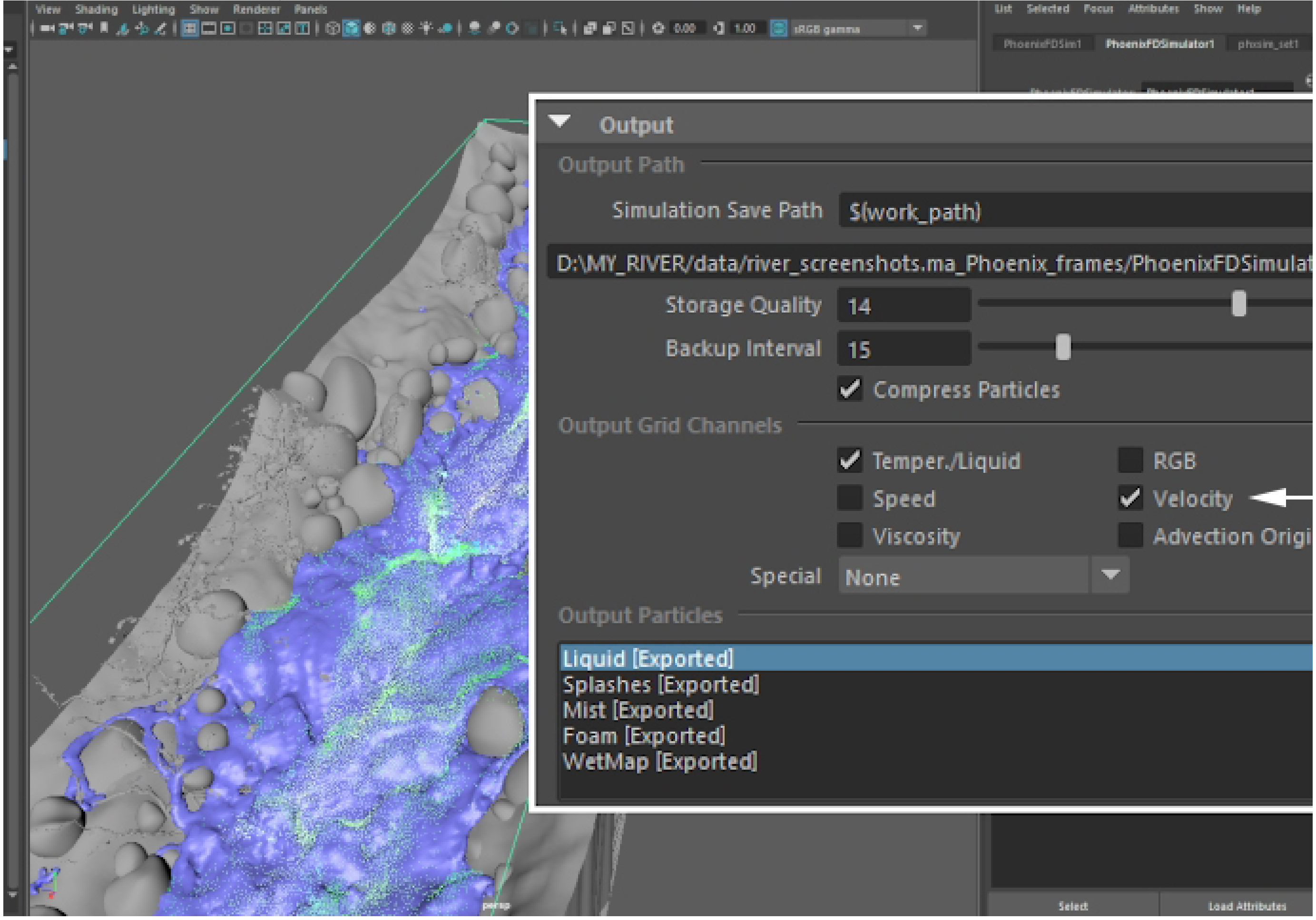
Pattern Strength 0.500

Pattern Radius (units) 10.000

Here's a preview of the Foam particles on their own:

At the moment, the Foam looks a bit chaotic. Once we reduce the Cell Size for the final simulation, the Foam particles will start clumping and producing interesting detail under the influence of the Patterns settings.

If you'd like to render the simulation with **Motion Blur**, enable the **Velocity Channel Export** from the **Output** rollout for both the **Grid** and the **Particle Systems**.



Output

Output Path

Simulation Save Path $(work_path)

D:\MY_RIVER/data/river_screenshots.ma_Phoenix_frames/PhoenixFDSimulat

Storage Quality 14

Backup Interval 15

☑ Compress Particles

Output Grid Channels

☑ Temper./Liquid □ RGB

□ Speed ☑ Velocity

□ Viscosity □ Advection Origi

Special None

Output Particles

Liquid [Exported]
Splashes [Exported]
Mist [Exported]
Foam [Exported]
WetMap [Exported]

Reduce the **Grid Cell Size** to **~2cm** for the final simulation and hit **Simulation Start**. Consider disabling the Mesh Preview and Particle Preview - reading the cache files and showing their contents in the Viewport becomes progressively slower the heavier a simulation becomes.

To the side is a preview of the final simulation.

We focus on the water material for the liquid, the Phoenix Particle Shader settings and an example V-Ray Blend Material setup using a Wetmap for the river stones.

Rendering

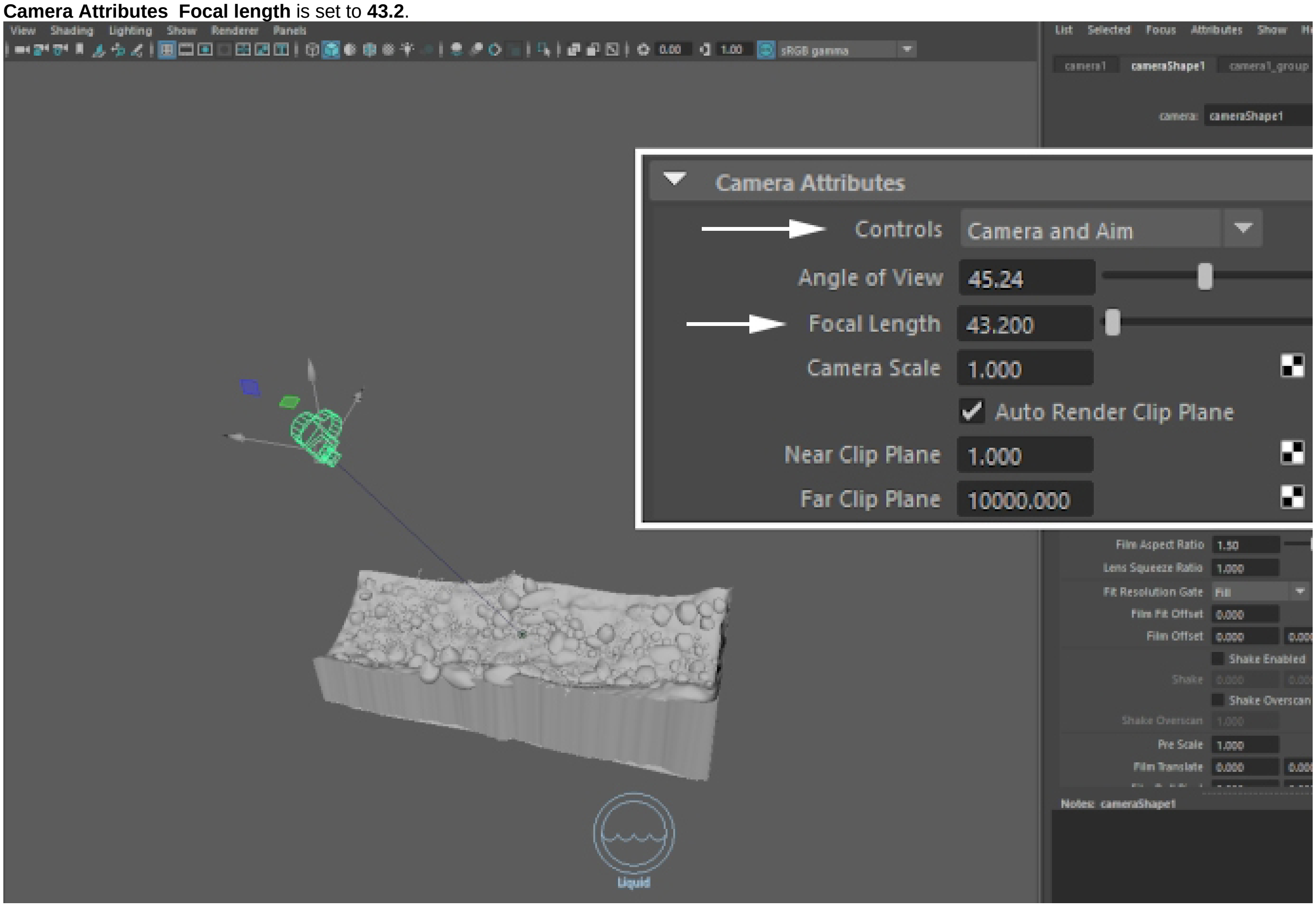
Let's start by adding a **V-Ray Sun&Sky** system and a **Camera** so the lighting setup is taken care of from the get-go.

Add a new camera by going to **Create Cameras Camera and Aim**.

Here are the exact transformations:

> **Target** Translate X/Y/Z: [-14.3, 566.646, -98.18]
> **Camera** Translate X/Y/Z: [434.102, 2463.551, -1014.203]

Camera Attributes Focal length is set to **43.2**.


View Shading Lighting Show Renderer Panels
0.00
1.00
sRGB gamma
List Selected Focus Attributes Show
camera1
cameraShape1
camera1_group
Liquid

camera: cameraShape1

Camera Attributes

Controls: Camera and Aim

Angle of View: 45.24

Focal Length: 43.200

Camera Scale: 1.000

☑ Auto Render Clip Plane

Near Clip Plane: 1.000

Far Clip Plane: 10000.000

Film Aspect Ratio: 1.50

Lens Squeeze Ratio: 1.000

Fit Resolution Gate: Fill

Film Fit Offset: 0.000

Film Offset: 0.000 0.000

□ Shake Enabled

Shake: 0.000 0.000

□ Shake Overscan

Shake Overscan: 1.000

Pre Scale: 1.000

Film Translate: 0.000 0.000

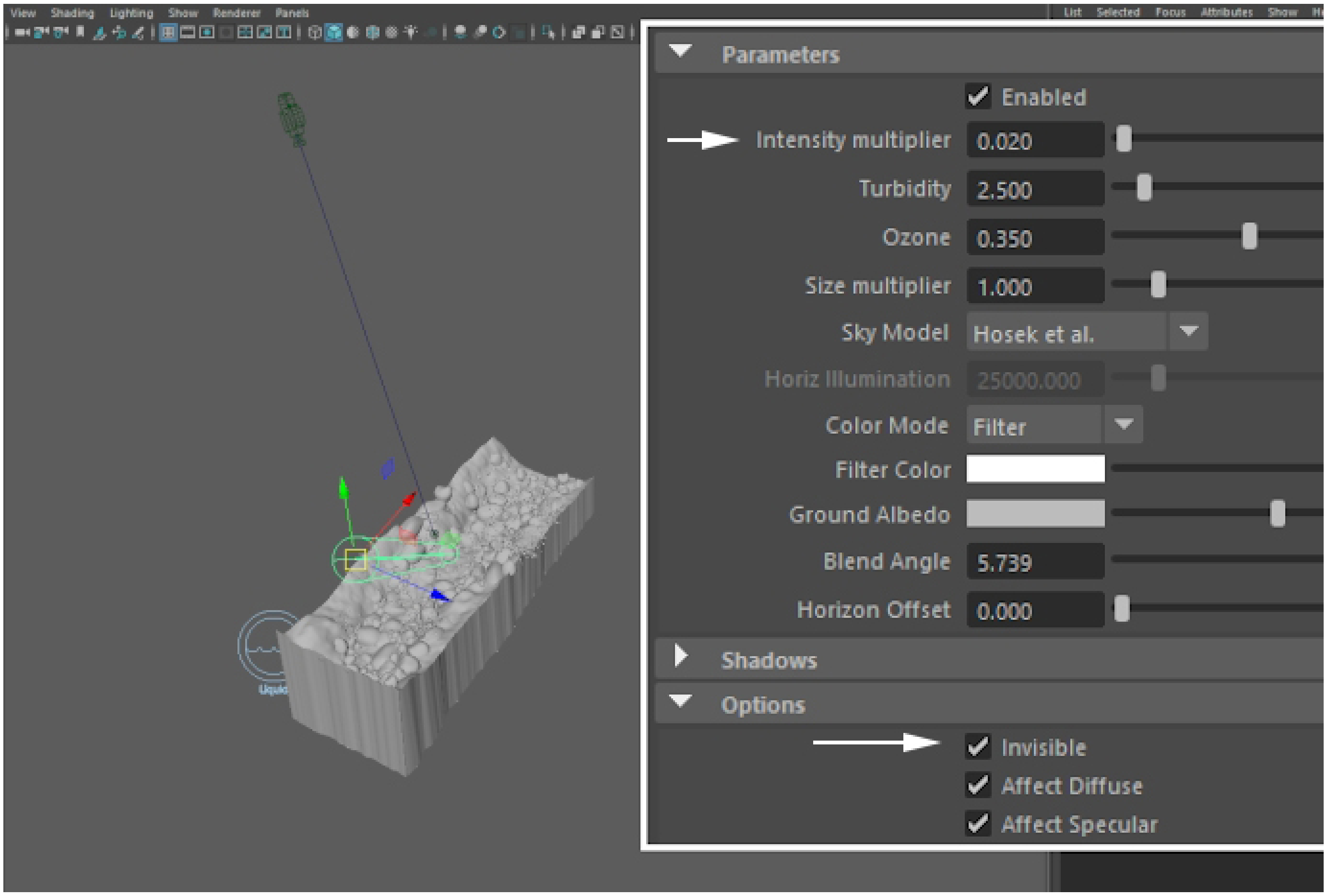
Notes: cameraShape1

Add a V-Ray Sun&Sky using the sun icon in the V-Ray toolbar. When prompted to choose whether a VRay Sky should be automatically created, select Yes. This will automatically set up the environment for you.

The exact transformations for the V-Ray Sun and its target are:

Target Translate X/Y/Z : [-31.599, 475.086, 0]
Sun Translate X/Y/Z: [-1974.218, 2511.039, 1133.8]

The **Invisible** toggle is **enabled** and the **Intensity Multiplier** reduced to **0.02**.



Parameters

☑ Enabled

Intensity multiplier 0.020

Turbidity 2.500

Ozone 0.350

Size multiplier 1.000

Sky Model Hosek et al.

Horiz Illumination 25000.000

Color Mode Filter

Filter Color

Ground Albedo

Blend Angle 5.739

Horizon Offset 0.000

Shadows

Options

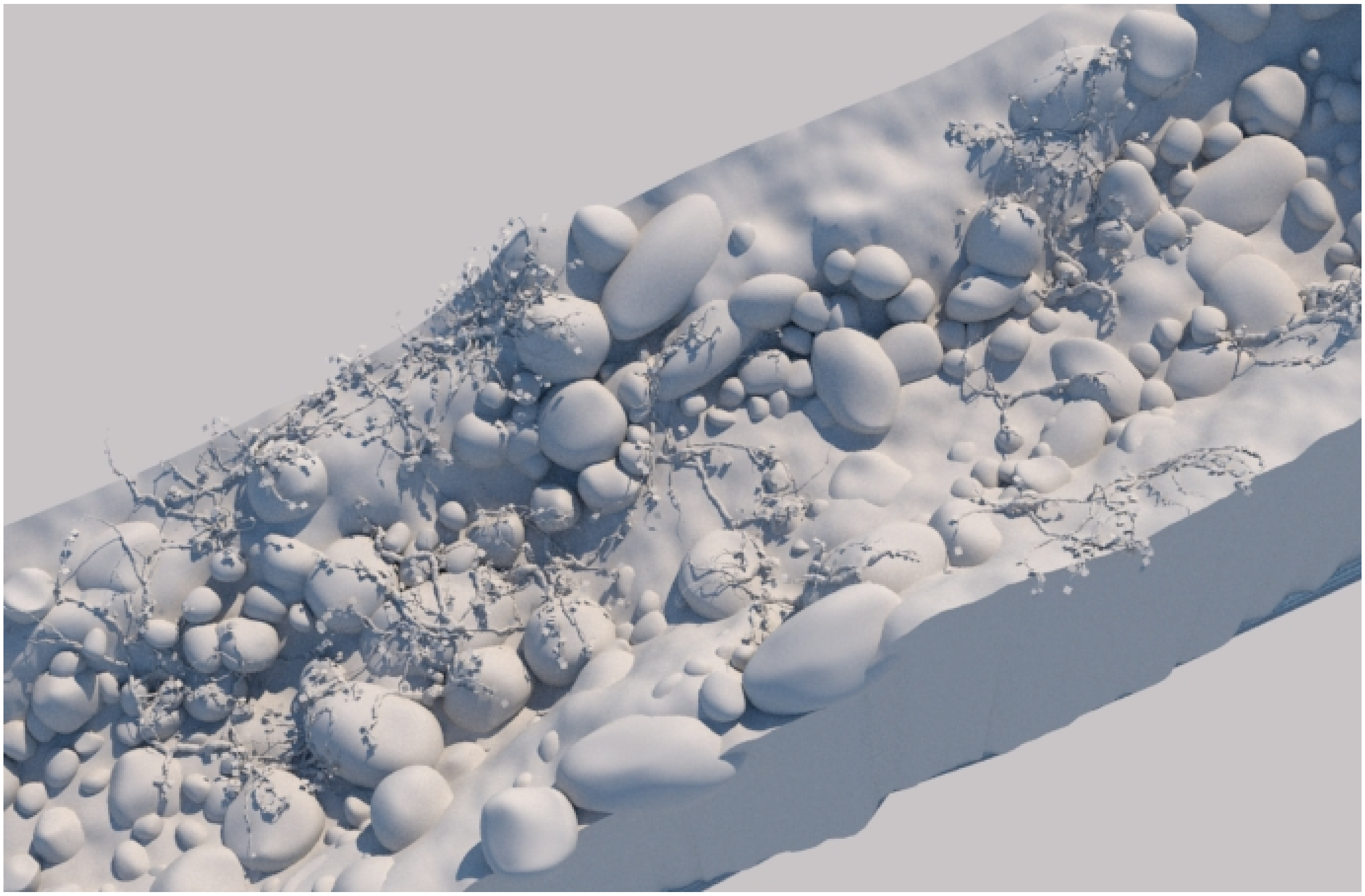
☑ Invisible

☑ Affect Diffuse

☑ Affect Specular

Here's a rendered image of the River Bed, River Stones and River Roots geometry.

The **Phoenix Simulator** and the two **Particle Shaders** for the **Foam** and **Splashes** are **hidden** to speed up the look-dev process.



Wetmap Setup for the River Stones

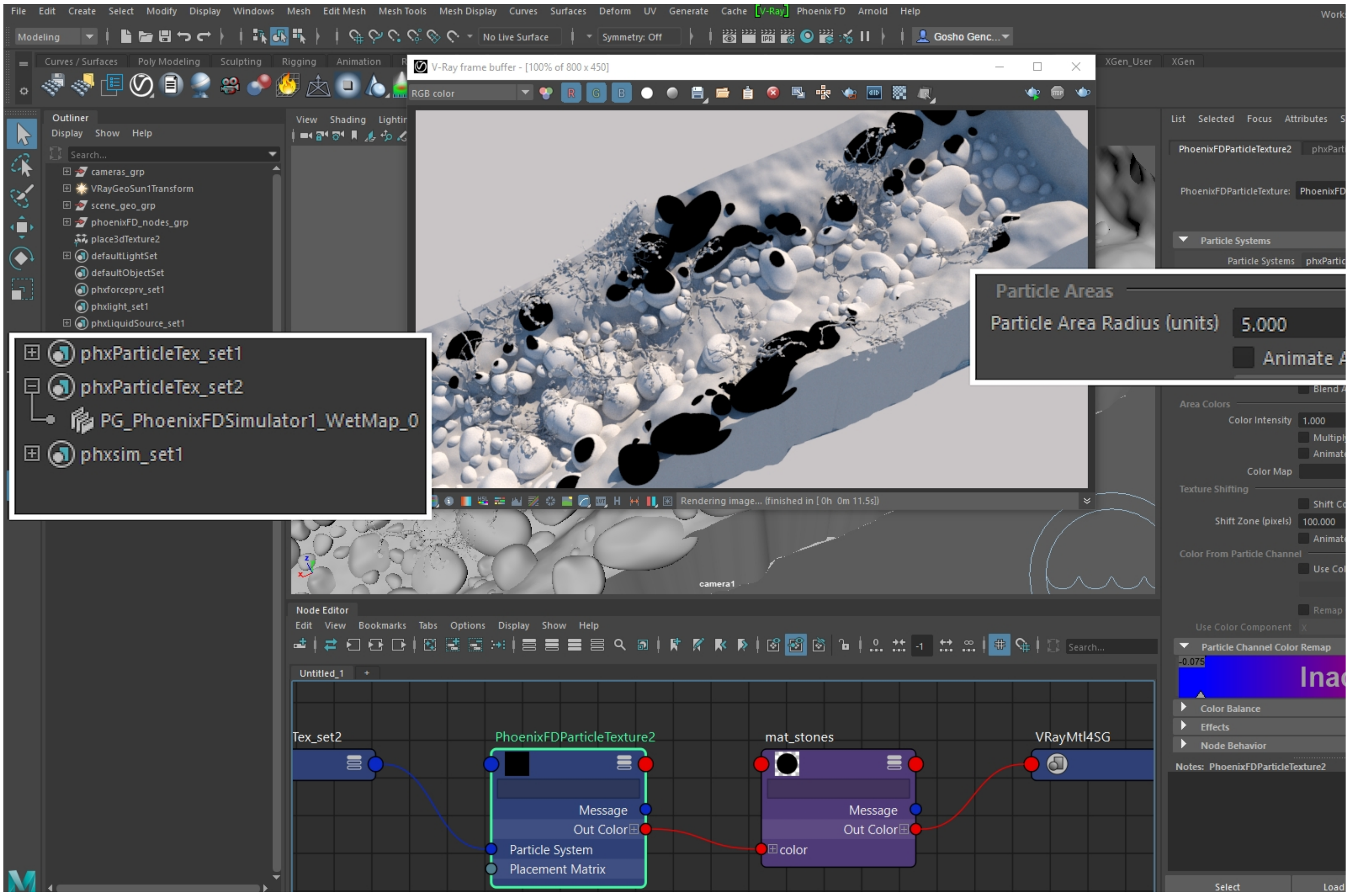
Assign a **V-Ray Material** to the **riverStones** geometry and plug a **Phoenix Particle Texture** in the **Diffuse** slot.

Add the **PG_PhoenixFDSimulator1_WetMap_0** node to the automatically generated **phx_particelTex_set** (recall that **Show DAG Objects Only** has to be **disabled** for the Phoenix Particle Group nodes to show up in the **Outliner**).

Set the **Particle Area Radius** to **5cm** to avoid noise artifacts in the generated black&white texture.

Hit render - only the stones which **interact with the liquid** should now render white.

Even if the Phoenix Simulator is hidden, the Particle Texture can access the WetMap particle group and use it at render time.



Rendering image... (finished in [0h 0m 11.5s])

Particle Areas

Particle Area Radius (units) 5.000

Color Intensity 1.000

Shift Zone (pixels) 100.000

Create a **V-Ray Blend Material** and **assign** it to the **riverStones** geometry.

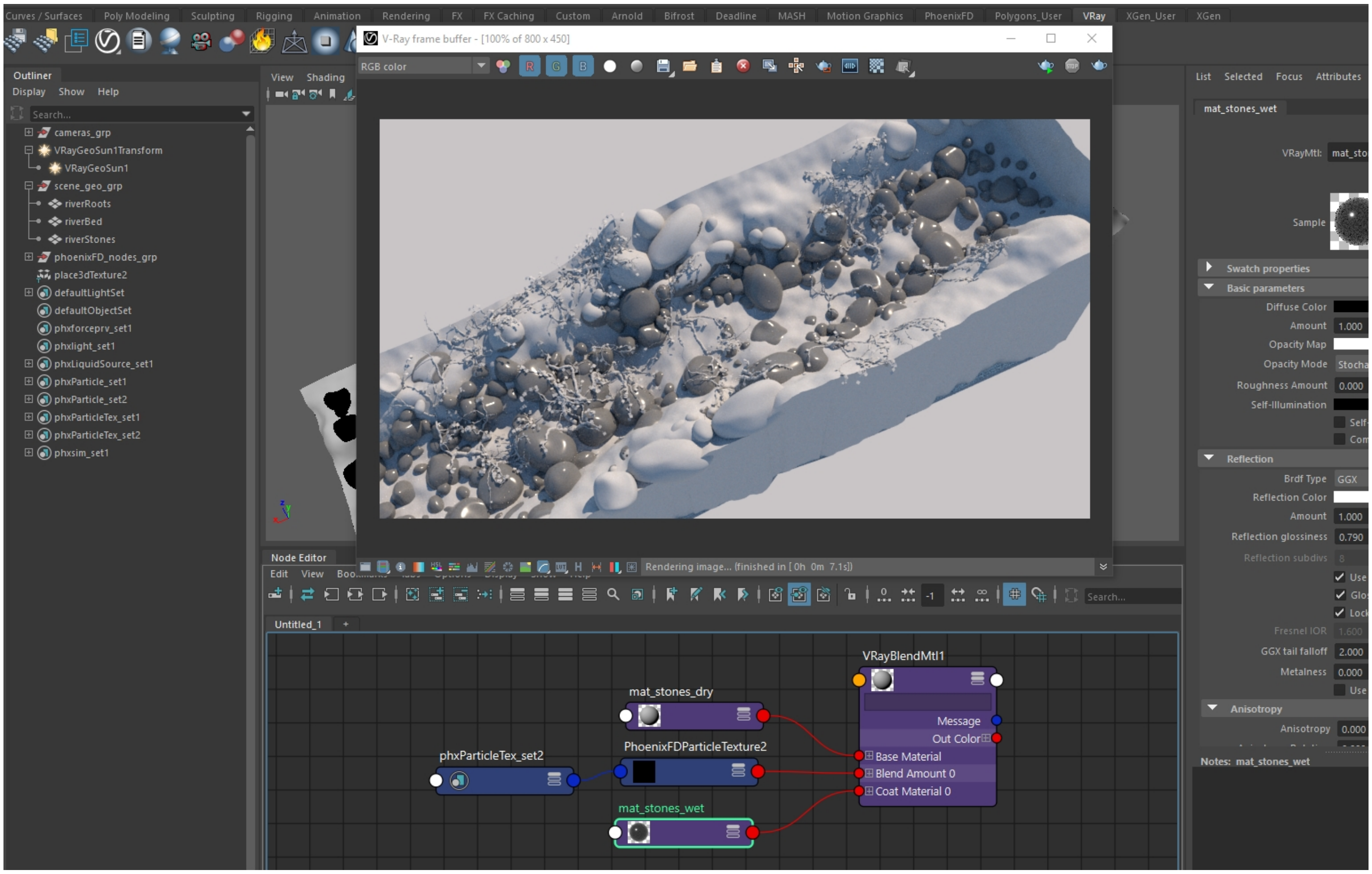
Plug the **Phoenix Particle Texture** you just created into the **Blend1** slot and **unplug** it from the **V-Ray Material**'s Diffuse slot.

Rename the **V-Ray material** to **mat_dry** or something along those lines and **pipe it into** the **Base slot** of the **Blend** material.

Add a **second V-Ray material**, rename it to **mat_wet** and **plug it** into the **Coat1** slot of the **Blend** material.

Here's how this works:

1. The Phoenix Particle Texture generates a black&white map (i.e. a mask).
2. The Blend parameter of the VRay Blend material expects a mask used to specify whether to use the Base material, or any of the Coat materials, or a blend between them).
3. When you plug the Phoenix Particle Texture in the Coat slot, the Blend material will use the Base material for the black areas of the texture and the Coat material for the white areas.
4. As can be seen from the image above, those correspond to the dry and the wet areas of the scene. All that is left to do is tweak the materials to get the desired appearance. For this example, the Reflect color of mat_wet is set to RGB [255, 255, 255] with a Glossiness of 0.65 so the wet stones are shiny.



Rendering image... (finished in [0h 0m 7.1s])

mat_stones_wet

VRayMtl: mat_sto

Sample

Diffuse Color ■
Amount 1.000
Opacity Map □
Opacity Mode Stocha
Roughness Amount 0.000
Self-Illumination ■
□ Self-
□ Com

Brdf Type GGX
Reflection Color □
Amount 1.000
Reflection glossiness 0.790
Reflection subdivs 8
☑ Use
☑ Glos
☑ Lock
Fresnel IOR 1.600
GGX tail falloff 2.000
Metalness 0.000
□ Use

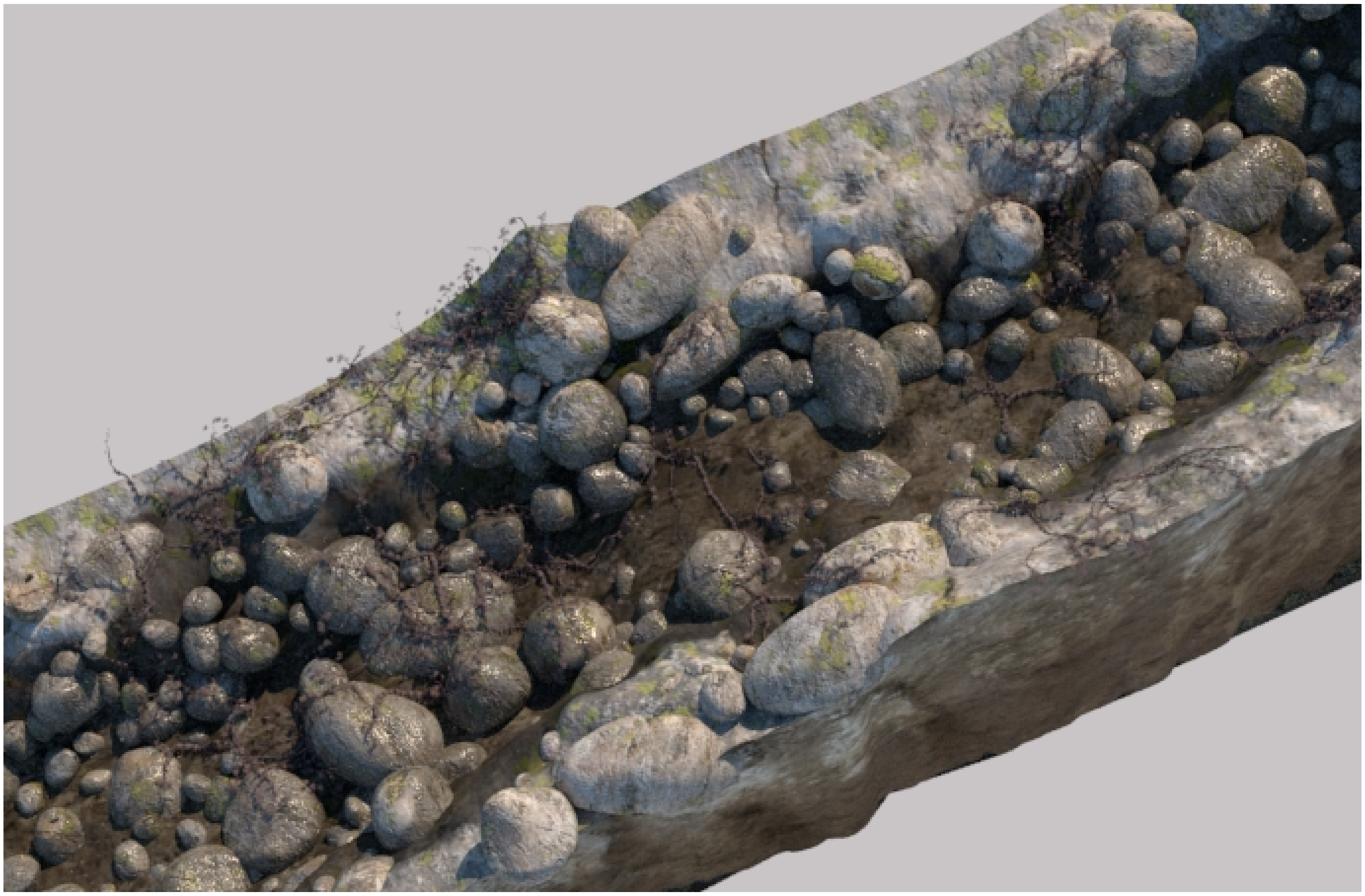
Anisotropy 0.000

Notes: mat_stones_wet

Here's a rendered image of all the materials from the example scene that ships with the tutorial files.

The exact same setup is used to drive the V-Ray Blend material using a Phoenix Particle Texture reading the WetMap particle group of the Simulator.

The textures used to generate this image are also provided.

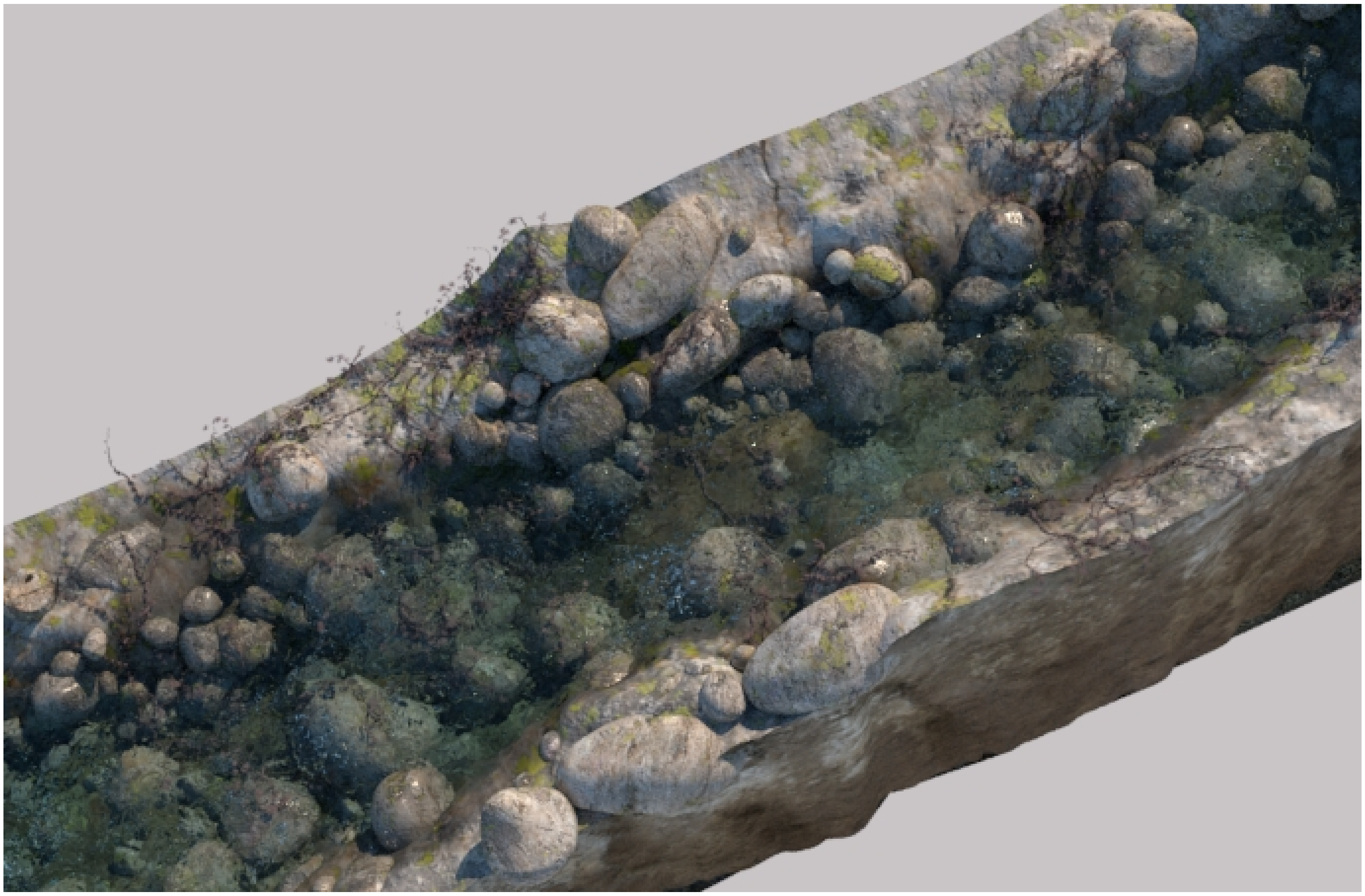


Water Material

Unhide the **Phoenix Simulator** and **assign** a new **V-Ray material** to it. **Rename** the material to **mat_water**.

Set the **Reflect** and **Refract** colors to **RGB [255, 255, 255]** and the **IOR** (Index of Refraction) to **1.33**. This is the physically-accurate IOR of water - you can use a search engine to look up the IOR of other materials in case your setup is different.

Set the **Fog color** to **RGB [246, 251, 251]** with a **Fog multiplier** of **0.2**. We use the Fog color to tint the water. In the real world, water is tinted because of the microscopic particulates it carries which refract the incoming light. Using the Fog options is an efficient way to emulate this without significantly increasing the render time.

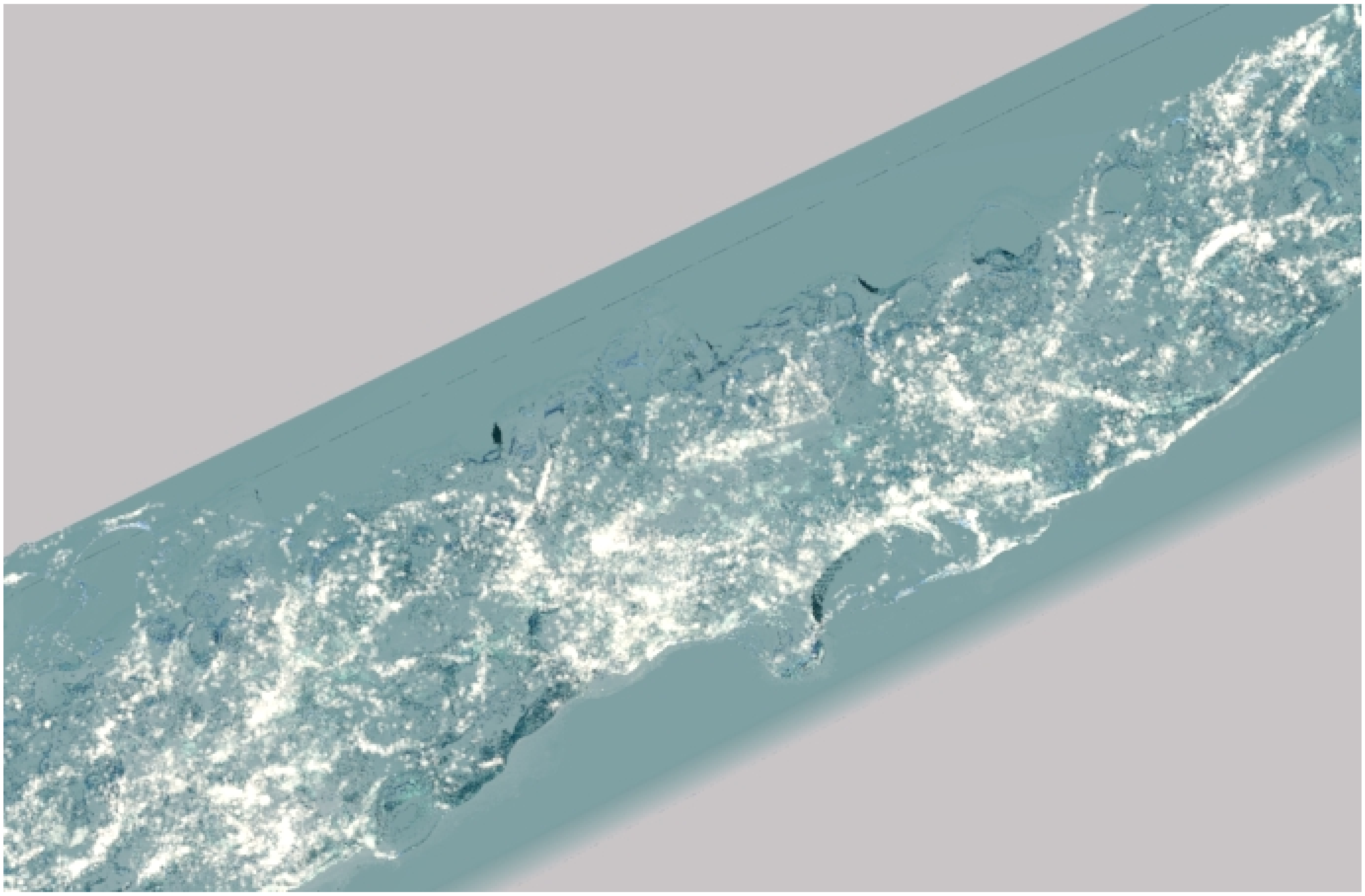


Foam and Splash Rendering

Here are the settings for the **Phoenix Particle Shader** used for the **Splash** particles:

Mode is set to **Points**, with a **Size Multiplier** of **2.0** to give the particles more volume. Feel free to experiment with the other Modes as well but keep in my this may increase your render times.

To speed up **Bucket** rendering, the **Light Cache Speedup** is set to **0.99**.

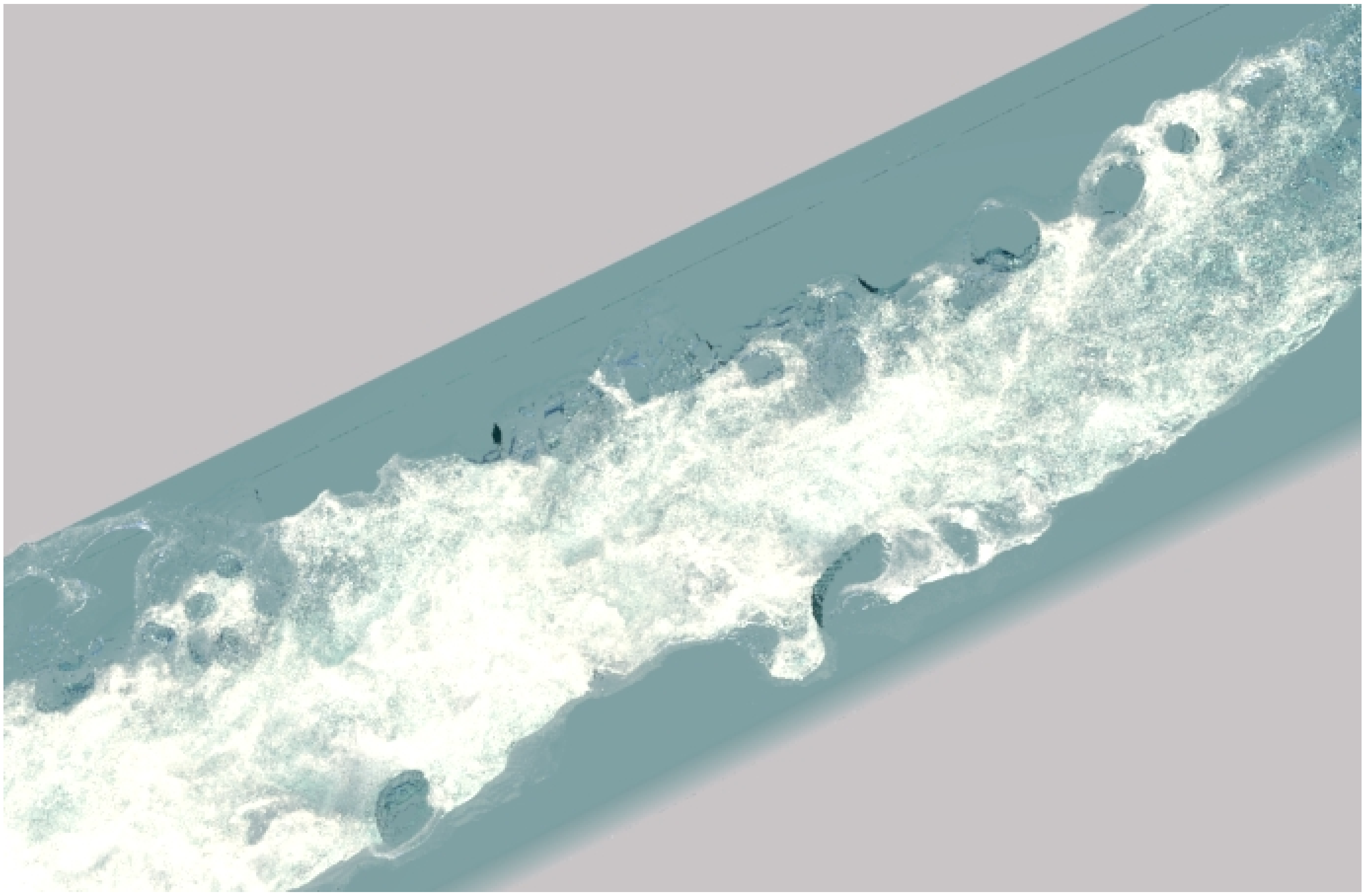


For the **Phoenix Particle Shader** used for the **Foam**, the following settings are used:

The Mode is set to **Bubbles**, with a **Light Cache Speedup** of **0.99**.

Under the Particle Shader **Cellular/Bubbles/Splash** rollout, the **Refractive Index** for the bubbles is set to **1.3** and the **Bounces** to **1**.

If the foam is too bright, as in the image to the right, reduce the **Diffuse Multiplier** to **0.9**.



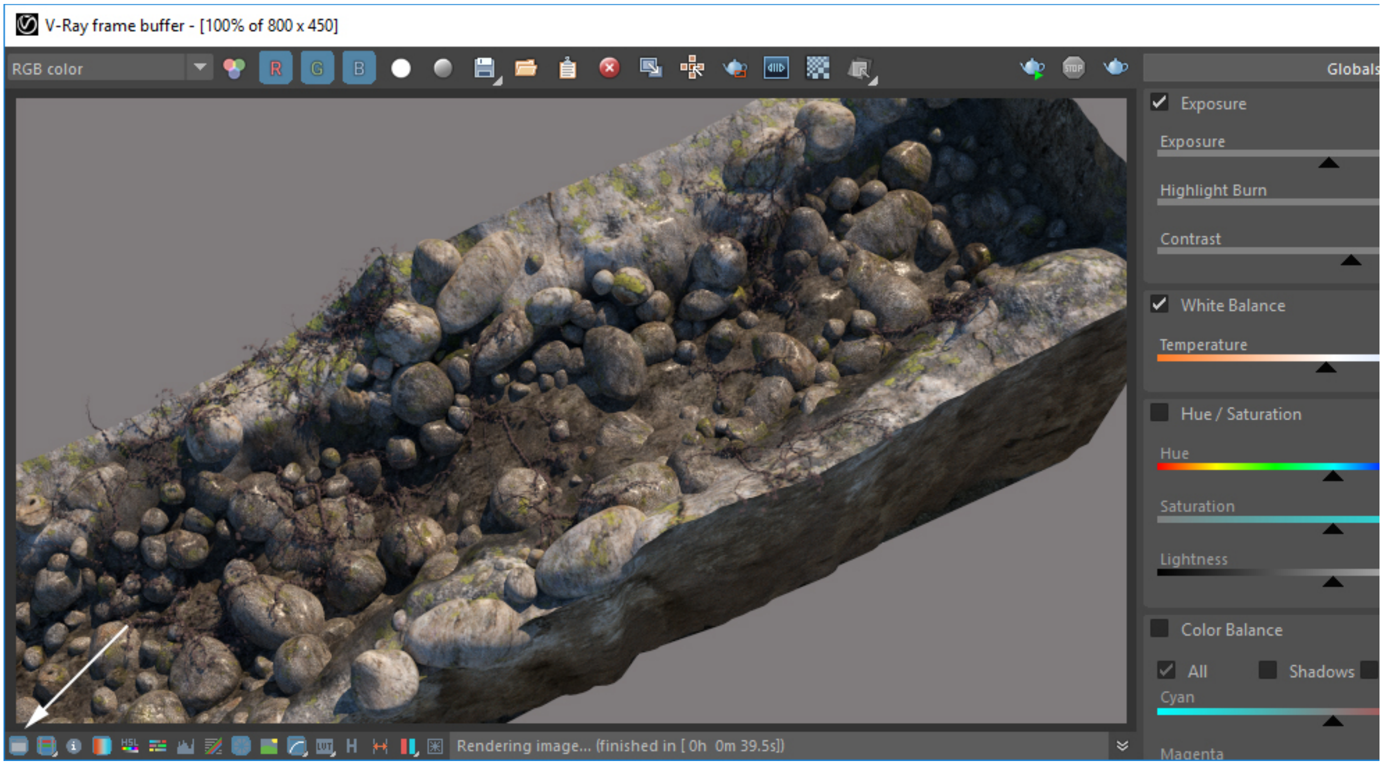
V-Ray Frame Buffer Color Corrections

Optionally, you may want to add the following color corrections:

Exposure is enabled and the following settings are applied:

> Exposure: -0.1
> Contrast: 0.1

White Balance is also enabled and the **Temperature** set to **6 350**.



V-Ray frame buffer - [100% of 800 x 450]

RGB color

Rendering image... (finished in [0h 0m 39.5s])

Globals

☑ Exposure

Exposure

Highlight Burn

Contrast

☑ White Balance

Temperature

□ Hue / Saturation

Hue

Saturation

Lightness

□ Color Balance

☑ All □ Shadows

Cyan

Magenta

Final image of all scene elements combined.