River Running Through It

Michael O'Brien
Pixar Animation Studios

Dave Hale
Pixar Animation Studios

Allen Hemberger
Pixar Animation Studios

Matthew Wong
Pixar Animation Studios

Figure 1: River in *Brave* ©Disney/Pixar All rights reserved.

For *Brave*, we created a naturalistic river in Scotland with several key story moments revolving around a waterfall and a calm play area over the course of 35 shots. It turned into the largest simulation project Pixar has attempted and allowed us to integrate high-detail physical simulation with artistically tuned details.

Base Simulation

We began with a volume simulation from Physbam of the entire set with 81 million voxels (each voxel was about $6cm^3$). The base simulation was authored in Houdini through a custom set of plugin nodes. We visualized Physbam's signed-distance-fields generated from the terrain and rocks dressed by our sets department as a collider. Other custom nodes allowed an artist to author and manage a Physbam simulation over multiple farm machines. The base simulation ran on 65 machines using OpenMPI to communicate boundary data during all phases of the simulation, not just frame boundaries.

We delivered the base simulation to layout and animation for timing and water depth. This simulation provided a contract between effects and upstream and downstream departments. This allowed the other departments to continue moving forward on shot work as we refined the water for each shot.

Windowed Simulation

Per shot, the river was simulated inside of the camera frustum colliding against animated characters at the highest resolution we deemed necessary. A typical shot was 160 million voxels at $0.5cm^3$. The per-shot simulation was bound to the base fluid at the boundaries and the divergence was injected or removed over the per-shot window to match the base simulation's volume. These conditions maintained the water level we gave to animation and avoided per-shot simulation tweaks of basic fluid properties. The windowed simulation provided much more detail around higher turbulent areas along with interactions of the fluid with the character's animation.

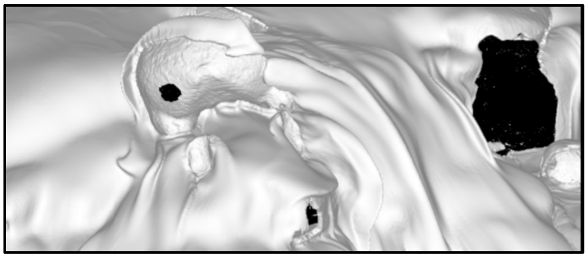

Figure 2: Windowed simulation ©Disney/Pixar All rights reserved.

The windowed simulation's surface volume was stitched into polygonal geometry as the substrate for the river's rendered geometry.

Secondary Simulation

The resulting volumes from Physbam drove a set of FLIP, 2D ripple, particle, and volume simulations to create waterfalls, character interaction, foam, and underwater debris. By having such high resolution data from the base simulation, we leveraged the details in the actual simulation as opposed to having to estimate them with techniques like injecting vorticity. The secondary simulations were highly art directed based on the director's feedback.

The majority of the simulations were a combination of FLIP work in conjunction with procedurally attenuated shading signals to provide visual details and mimic different water types.

Shading

All of the disparate data was rendered by creating a specific set of water shaders that aided us in visually transitioning from mesh to particle data.

For the surface, two sets of coordinates were advected using the windowed velocities. These coordinates were reset every 50 frames in alternating intervals. Both coordinate frames were used to attach a Tessendorf displacement over the surface of the river for added detail. The two patterns were blended over the 50 frame interval. At each point in the mesh, the shear of the fluid space was also computed. High shear areas in coordinate space were usually indicative of the boundary between two different flow speeds. If the amount of shear was too high, the Tessendorf pattern would be dampened to avoid banding artifacts.

Figure 3: Shading detail ©Disney/Pixar All rights reserved.

The particle shader supported our full refraction and reflection illumination model. The necessary surface parameters were tuned by artists per shot with a wide variety of techniques. For more spray, randomness was applied to most parameters. For more sheeting, the particles were meshed into an implicit surface, signals were computed on the mesh, and then transferred back to the particles for rendering.

Result

The result was an artistically tuned suite of shots that all rely on volume data that is coherent at multiple scales.