What Went Right

Ψ Global Post-Processing Chain

Initially, each level had an individual post-processing volume but there were no global settings. After seeing some of the more powerful effects in action, I researched how to get things like ambient occlusion into the game. These can't be in a post-processing volume – instead, they need to be part of a post-process chain. I researched post-process chains on the Unreal Developer Network (UDN), and used the knowledge to set up global post-processing, including ambient occlusion, in our game. This gave us very strong shadows with relatively little overhead, and allowed more scene coloring through individual post-processing volumes as the global post-processing helped keep shadows black.

Ψ Per-Level Lighting/Coloring

As my major role for the project was setting up lighting across the game, I wanted to have each level have a palette that clearly reflected the primary element and helped reinforce the emotions we wanted the players to feel. For example, the water level uses primarily blue and purple hues, with orange torches and natural yellow light to contrast while the boss level uses one strong dominant light with light rays, positioned behind the boss to lend more grandeur to the scene with some smaller colored lights to draw players to various puzzle elements in the environment. This method of doing per-level lighting and coloring made each level feel unique and allowed the team to get extra mileage out of a limited asset pool.

Ψ Player Pathing

I used three primary tools when setting up lights to enhance player pathing: in darker areas I kept the critical path mostly lit, in brighter areas I used contrasting lights to highlight exits or important puzzle objects, and throughout the fire level I used shadows to draw lines between puzzle elements. The last was necessary because throughout the fire level, fires were a key puzzle element as well as the dominant environment light. Because of this, there was less opportunity to use contrasting light to highlight important elements and exits. To counteract this, I began attempting to draw rough shadow lines by placing meshes – such as pillars – between the fire source and the intended target or room exit. It was a subtle way to help players find their way through the environment and prevent them from becoming lost, but it worked well.

What Went Wrong

Ψ Light Function Performance

Part of the solution to get light from flames flickering appropriately was creating Light Functions for the lights, which created a very realistic flickering effect but also taxed performance. Early in the project, when these lights first went in, this performance hit was barely noticeable. However, as the end of the project approached, we were consistently getting big performance hits and could not figure out why – it didn't seem like it was one thing that got added, performance just got worse as all the completed assets were in. After figuring out that the performance was markedly worse in areas with several fires, I made a pass to ensure that only one light had a Light Function in any given scene. This immediately boosted the game's performance, and was an integral part of a coordinated optimization pass. Additionally, by smartly placing the one flickering light, it was able to simulate flickering from multiple fires, so the loss in lighting quality was minimal. I learned that, with dynamic lights especially, a light touch works far better than a heavy hand.

Ψ The Gondola

During the initial design phase of the project, I worked with a partner to design the final non-boss level. This level combined all of the elements, and served as a mastery test for the player to lead in to the boss. During the level, the player rides a large gondola through a mountain, solving puzzles as it moves, racing alongside it, and in rooms to get it restarted. Conceptually, the Gondola would serve as a home base of sorts, and the player would develop an attachment to it over time. In practice, many of the moving puzzles did not playtest well – players felt unfairly pressured by the harsh fail conditions (as the gondola would fall from the tracks) and at-times unclear objectives. Additionally, the Gondola sequence required a large number of short Matinee sequences that cost many hours, and required significant tweaking for balance. In the end, the level turned out well, but these underlying problems meant the level required significant rework.

Ψ Level Transitions

With each level having a unique lighting palette and post-processing settings, and the game taking place in a contiguous world, transitions between the post-processing settings was important. I attempted to have a slow fade in/out time on each volume, which did create a somewhat appropriate fade between each. However, as it was a third person game, the camera flew several feet behind the player. This meant that, even after the player crossed into a new level the camera may still be in the previous. This, combined with the camera clipping between volumes, meant that these transitions were not as smooth as was intended. Unfortunately, time pressures shifted my focus elsewhere and I was unable to solve this problem during the project's development – a short time after conclusion I had the idea to stream out the old volume and stream in the new, which would mean each transition could only ever happen once.