Allie Lavacek

Gameplay Engineer
Team 4 | Blizzard
University of Michigan
Computer Science
Graduating Dec 2024

Gameplay with OW2

My 3 month internship experience on OW2

During my internship at Blizzard, I was able to...

- Onboarding - Prototype 3 hero abilities
  - Added Debug Draw Triangle Node to TED
- Allow designers to more easily add charges to hero abilities
- Nerf junkrat (you're welcome)
  - Allow a projectile's size to change on ricochet
- Output a list of points where a shockwave (Earthshatter, Overrun, etc.) ends or hits something
  - To be used for effects
- Implement Shockwave Rewind
  - After a shockwave finishes, it can now rewind back

Key Learnings

- **TED and Statescript**
- **T4 Codebase's structure**
  Specifically pertaining to Abilities and WeaponVolleys
- **Assessing code perf**

What I Learned

TED and Statescript

TED is the engine we use on T4 and Statescript is the visual scripting language. I got a better understanding of why Statescript being Stateful is beneficial for the game and why different games may require engines with different architecture.

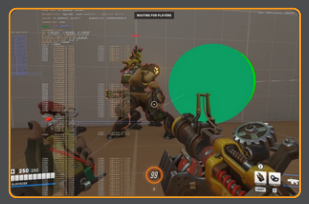
I used both to implement the 3 prototype hero abilities for my onboarding task.

T4 Codebase

After my onboarding task, my work shifted from implementing something solely with Statescript to instead making code changes to modify Statescript nodes. This required understanding stu files and generated code, as well as understanding how certain nodes are implemented.

I worked inside of the Weapon Volley scripts, responsible for creating and handling Projectiles, and the Ability script- which both contain thousands of lines of code. I found that at the beginning of each of my new tasks, I would require a day or two to simply read the code, and understand how everything worked together. Moreover, before writing any code, I learned that its essential to understand how my changes should fit into existing patterns.

Additionally, my tasks pushed me to better understand what order scripts are being called in, communication between the server and client, various memory allocation patterns, and edge cases to consider for gameplay code.



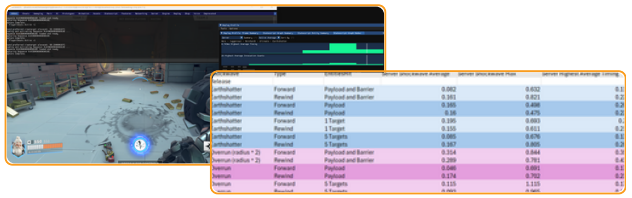
For example, for my projectile ricochet task, I came across a bug where if a projectile hit a wall, the server and client would both shrinks its size, and then the server would feed the client outdated information which caused the client's size to be larger for a couple frames, leading to occasional mispredicted effects being spawned. Understanding communication between the server and client was essential for me to tackle this issue and better understand what should and should not be synced between the server and client.

Understanding of these concepts also helped me approach problems quicker, such as knowing where to look to spawn effects for Shockwave rewind, or what information needed to be communicated through MirrorData.

Assessing Code Perf

A huge part of any programmers job is ensuring their code runs efficiently. Therefore, I had to learn various ways to assess the performance of my code, and determine if or where improvements needed to be made.

The main tool I used was our Replay Profiler. This tool allows us to record gameplay, and play it back. Once playing it back, we are able to see how many milliseconds on average each Statescript graph took per frame, and how many milliseconds on average and at max a specific node took, on both the client and server. This was specifically useful for my two changes for Shockwave, as Shockwave is already a fairly expensive ability to perform.



Additionally, I explored tools that allowed me to see what the most expensive functions of my code were.

Confluence Pages

- https://confluence.blizzard.com/display/T4/Ability+Node+Charge+Field+Setup
- https://confluence.blizzard.com/display/T4/New+Shockwave+Endpoints
- https://confluence.blizzard.com/display/T4/Shockwave+Rewind

If I had another 3 months....

- Potentially work on non-hero related gameplay tasks
- Tackle bug fixing tasks
- Get a better understanding of how systems pertaining to other areas of Gameplay work such as maps and modes, animation, and mythic skins/weapons

Conclusion

I am thrilled with the learning opportunities that I was provided with during this internship. I am certain that the skills I developed over the last 3 months will serve as a solid foundation for the career I have ahead of me!