THE PENNSYLVANIA STATE UNIVERSITY
SCHREYER HONORS COLLEGE

COLLEGE OF INFORMATION SCIENCES AND TECHNOLOGY

AN EMPIRICAL APPROACH TO ITERATION IN LEVEL DESIGN

PATRICK MULHOLLAND
Spring 2011

A thesis
submitted in partial fulfillment
of the requirements
for a baccalaureate degree
in Information Sciences and Technology
with honors in Information Sciences and Technology

Reviewed and approved* by the following:

Andrea Tapia
Associate Professor of Information Sciences and Technology
Thesis Supervisor / Honors Advisor

Barton Pursel
Research Associate, Schreyer Institute for Teaching Excellence
Thesis Reader / Advisor

* Signatures are on file in the Schreyer Honors College.

ABSTRACT

The best practices behind designing a map, or level, have been debated within game design circles since the creation of the field. This thesis encompasses a review and extrapolation upon established level design techniques used within the industry. Methods of level improvement covered by this paper will focus on the analysis of metrics useful during playtesting. The author's own interpretation, application and furthering upon of current techniques will be explored through an in-depth case study. The study, the subject of which has been published in a professional video game title, demonstrates the quantifiable results of intelligent iterative changes throughout the development cycle, suggesting a successful process of empirical gameplay improvement.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS

I could not have gotten to the point I currently stand as a level designer, let alone completed this thesis, without the following:

Ryan Mannion and for his extensive playtesting, feedback, and support with data mining;

Arthur Bobany, my partner, modeler and author of the payload model in frontier, “Lil’ ChewChew”;

The tf2maps.net staff and members, especially my friends and peers Tim Johnson, Wade Fabry, Eric Wong, and Joachim Holmer, who have provided years of solid criticism and assistance on many levels;

Corey Peters, Eric Smith, Robin Walker and other employees of Valve for their feedback and inclusion of entity requests into Team Fortress 2.

I owe additional thanks to those here at our school that have seen me through to this point academically:

My former honors advisor, Dr. Stan Supon;

My current honors advisor and my thesis supervisor, Dr. Andrea Tapia;

My thesis reader, Dr. Bart Pursel.

The heatmaps generated for this thesis were made with the open-source tool PsychoStats, which can be downloaded at http://www.psychostats.com/

Chapter 1

Introduction

Levels—otherwise known as maps, worlds, or environments—serve as the fields upon which video games play out. They come in the form of dungeons, mountains, space stations: whatever they need be for a given game. The thoughts behind the construction of these game spaces, from little more than concept to an immersive 3D experience, are known collectively as level design. Proper level design can serve in many ways to enhance a game's impression on consumers: levels can help establish the setting, atmosphere and mood of a game. They may also help convey story elements, or apply pacing adjustments to the player experience. Appropriate level design is an important factor in the full utilization of other game assets, whether those be programming or art resources. Lastly, and perhaps most importantly, level design determines the ways in which players interact with core gameplay mechanics.

While games do indeed stand to benefit in the above ways through solid level design choices, making those choices and correctly implementing them forms the artistic challenge behind the field's application. Most level design resources today focus on the narrative, forgoing details of iteration for high-level theoretical analysis. This paper, conversely, will focus on directly observable gameplay-oriented design improvements. The goals of these approaches to drafting and iteration are to induce more qualitatively positive reactions from players by correctly interpreting and responding to quantitative playtesting data.

The individual chapters ahead will cover the case study's level design process from concept to release, providing concrete examples of empirical improvement. The takeaway for the

readers of this paper will be an understanding of how a level may be manipulated based on quantitative data to maximize the player experience in terms of interest-engagement and fairness.

If unknown terminology is encountered, **please refer to Appendix A** for a defined list of relevant vocabulary.

Chapter 2

Case Study Primer

Before launching into the particulars of the study, it is pertinent to provide contextual reference regarding it for ease of understanding. For readers already familiar with the game Team Fortress 2 it may be recommended to skip this chapter's introduction to the game and the engine on which it runs.

2.1: The Source Engine

The Source Engine was developed by Valve Corporation out of the GoldSrc engine that ran previous titles such as Half-Life and Counter-Strike; its roots may be traced all the way back to the Doom engine. Today, it boasts an impressive collection of games, for the most part developed by Valve itself: Half-Life 2, Counter-Strike: Source, Left 4 Dead, Portal, and Team Fortress 2 rank among them.

Source levels consist of additive geometry, meaning that level designers working in the engine create within a void. This runs in contrast to subtractive engines like Unreal; whereas Source functions by sealing a map from the vacuum surrounding it, Unreal maps are tunneled out of solid space. The most relevant effect of this is that Source mappers have to consciously block off holes, or leaks, from exposing their sequestered environments to the void.

Source maps are stored in the proprietary Valve Map Format, taking the suffix .vmf. Once compiled, the maps take the file type .bsp, or Binary Space Partitioning. This name references the basic method by which visibility is determined within Source. In order to determine what surfaces need not be rendered, Source approaches the problem by, during

compile, bisecting the game world along existing geometric planes into smaller chunks of space. This process is repeated until the entire map has been subdivided into convex spaces, at which point the compile determines which spaces are visible from a given space. The engine then knows that if a player is currently in that space, it only has to render those predetermined areas. As a result, a player's computer endures less stress during rendering at the cost of longer compile times on the developer's end. It does however put limits on a level designer's choices, as the creation of geometry must be aimed towards supporting this culling method.

As Source is, for the most part, a statically lit engine, any lighting within the map must be baked after visibility calculations have been made. The compiler accomplishes this by sending out imaginary light rays from each source in the map and bouncing these against world geometry. Each bounce leaves a computed amount of light on the receiving surface until the ray is exhausted.

Blocks of geometry inside a Source level are referred to as "brushes" and form the vast majority of a given level's contents. Each brush is defined by a series of vertices, which themselves describe planes that form the brush's faces. Faces contain information about their individually assigned materials—substances consisting of the textures and properties that will be applied to the face itself. Quadrilateral faces may be split into displacements, surfaces with numerous individual vertices that can be then manipulated in the 3D coordinate plane, essentially 'displacing' the face into terrain or curved surfaces.

As Source is at its base a C++ application, game mechanics are all defined in an object-oriented manner. Within a Source map, object classes are referred to as "entities," and may typically be one of two different types. Point-based entities' positions are just that; they inhabit spaces dictated by their coordinates in 3D. Examples include the players' start positions, all lighting sources, goals, pickups (health, weapons, ammo etc.) and the props that form the most

finely-detailed parts of a map. Brush entities are related, or ‘tied’, to pre-existing brushes; these range from doors, to illusionary brushes, to triggerable areas, to breakables and more.

2.2: Team Fortress 2

Team Fortress 2 (or TF2) is a Source-based game released by Valve in 2007. Its predecessors date back to the 1996 Quake modification, Team Fortress; however, the basic idea behind the game has influenced and appeared in many titles throughout the years. In most basic terms, TF2 can be classified as a solely-multiplayer goal-based first-person shooter. A server must host every TF2 game on a map, with clients joining onto the server to play. In each map, two teams of players compete against each other over an objective. Games can range from as low as 1v1 to 32 player brawls, 16 on each side. Players may take the role of any of 9 different character classes, distinguished from one another by appearance, personality, weapon set and ability.

Of particular note is that some of these classes are inherently more suitable for defensive or offensive roles than others. For example, the Sniper class functions best when far away from a battle, and the Engineer class relies on stationary constructions of the player’s building, giving them ample ability to hold down an objective. The fast-moving Scout class, on the other hand, does most damage at close-range, and paired with its ability to gain control of objectives faster than other classes makes him an offensive class. A healing class, the Medic, must pair up with another player in a symbiotic relationship in order to contribute to the team. Because of this class variety TF2 requires of its players more cooperation and teamwork to accomplish in-game goals than most first-person shooters. The ability to communicate via built-in commands, text-based chat or by VOIP helps to facilitate teamplay.

Players in TF2 have amounts of health respective to their chosen class. When this health is depleted (by taking damage, falling, being lit on fire etc.) the player dies and is forced out of the game until the next respawn wave. This health can be replenished either by returning to a spawn location, by a health pickup, or by being healed by a medic. Players also have ammo for their weapons, which also fuels the spy's invisibility ability and the engineer's ability to build defensive sentries. Ammo may be replenished either by ammo pickups, returning to spawn, or standing near the engineer's "dispenser" building.

Maps are defined centrally by their objective type. While many permutations exist within the game, the core gamemodes of TF2 remain capture-the-flag, control point and payload. Capture-the-flag is a classic among multiplayer first-person shooters: each team has a 'flag' to defend, while attempting to touch the other team's flag and bring it back to their own. Control point maps consist of several areas that must be stood upon, uncontested, by one team in order to gain 'control' of that area's point. Once all control points (CPs) are owned by a team on a map, that team wins the game. Some maps have all CPs initially owned by one side, creating an assault/defense situation, while others have several linear CPs setup in tug-of-war fashion.

2.3: The Payload Gamemode

Payload is entirely unique to TF2: an objective must be escorted, uncontested, along a predetermined path to a goal in order to capture each CP. This objective, titled the Payload itself, dispenses health and ammo to offensive team members close enough to escort and 'push' it forward. As more uncontested offensive members push, the cart's speed ramps up (to a given point,) encouraging cooperation; if untouched by offense for 30 seconds, it begins to roll backwards to the previous CP, at which point it stops. BLU, the offensive team, races against the round timer's countdown to push it to each CP, rewarded with extra time as they capture each CP

in turn. Should they fail to reach the next CP before the countdown expires, the defending team RED receives the round's victory; however, if they capture every CP the Payload explodes and BLU wins for that round. **Please see Appendix D** for an example of a Payload track layout. The traditional Payload takes the form of a cart carrying a large bomb, together the approximate size of two players standing side-by-side.

Payload and other assault/defense gamemodes contain two periods at each end of the playing experience that differ from the rest of the maps' gameplay. The first period is called "Setup", a time during which BLU cannot leave its respective spawn. During this period, RED players can travel to the frontline of the map, usually directly outside BLU's first spawning area. Engineers use this time to create defensive sentries, and medics make sure everyone on their team is fully healed. This creates a period of high-intensity, but short-lived, battle as soon as Setup ends. The second time period occurs when one team has won for the round—this "Humiliation" period totally disables the losing team's players and grants the winners greater firepower for its duration. It serves as a bonus incentive for winning as a team rather than as an individual, a challenge-free killing spree for the victors.

TF2 is an interesting study for several reasons. The existence of a strong playerbase, combined with the scalability of play in terms of player count, provide a solid pool from which to draw playtesters. Its well documented server log structure makes visible each discreet game action, leading to quantified gameplay reports. The engine also supports flattened top-down rendering for the creation of heatmaps, etc. As gameplay elements are well established in the canon of existing maps, even in completely unfamiliar territory players will be able to pick up on goal clues and proceed in a generally objective-oriented fashion. Taken together these qualities allow for easy demonstration of core level design practices in a controlled environment.

Chapter 3

Case Study: Frontier

3.1: Inception

Valve's unique payload gamemode immediately gained huge popularity among the TF2 fanbase: to this day, the first two released maps within the mode remain at 3rd and 4th most played, respectively, among the 40-odd maps currently included in the game. The gamemode was not without its flaws, though. The cart, around which offensive players have to gather in order to progress, forces individual team members to sacrifice their autonomy and any locational advantage they might have in order to progress through the map. The slow-moving cart's capped speed was originally intended to allow defenders time to regroup and fight back in between spawn waves of reinforcements. But the gameplay that emerged more closely resembled a pattern of the offensive team fighting forward to the next choke point, and then subsequently bringing up the cart in a slow and predictably boring fashion.

The first of my design goals in creating the map later-to-be-called Frontier revolved around these issues. I decided to follow the format of Valve's second released payload map, Badwater. Badwater is a single stage map containing 4 capture points along its payload track. This is in direct contrast to the original payload map, Goldrush: Goldrush contains 6 points, but they are evenly distributed across 3 stages. As each stage resets players to their respective spawning locations, the multi-stage nature of Goldrush only exacerbated that constantly-halted gameplay. Another effect of Goldrush's greater length was the lengthening of an average round's time to completion on the map, testing player patience. As a result less of the total map length is

experienced per-round, minimizing the developing effort : user impact ratio for the map. I found Badwater to be clearly preferable as a framework, in terms of gameplay, upon which to build. Frontier would also follow the 4-CP format.

To combat the low appeal of the cart as a rally point, and as position of strategic value and benefit to individual players, I wanted to have a more interesting, offense-oriented and faster platform. Having played an amusing gimmick map involving a high-speed train platform on a closed circuit, I had already witnessed the fun players could have jumping on a platform and riding it while fighting. In fact, this was already possible to a degree on the top of the bomb-like cart of the payload gamemode, but the available space was very small and offered no cover. The second failing of the gamemode—the jarring, push-then-pull strategy encouraged by precedent—I decided could also be addressed by ramping up the cart's max speed to a point where progressing on foot would be only marginally quicker.

With these points to be addressed in mind I started sketching out concepts for a train-engine / train flatbed cart. The train would be faster than the Valve payload, provide more cover behind its imposing engine, and allow players to dodge while riding upon the significantly wider flatbed: while the original might support 2 players on top, Frontier's train could hold an entire team. In addition, I thought the train might gain the ability to run over and kill players in front of it, given enough offensive members on the flatbed: more players would lead to greater speed and greater damage capacity. A mockup using simple simple brush/entity work was drafted to determine scale. Immediately it became clear that the existing flatbed prop would be too large to take any sharp turns, and so it was replaced with a temporary brushwork solution. **Please see Appendix E** for pictures referencing the development of Frontier's unique cart.

Having the mockup helped me to form my vision for the level, of a supply line coursing through buildings and up hills on its way to an enormous surface mine—a setpiece that would later help to define the map. Paper prototypes were drawn up and tossed out until I had enough of

a workable concept to begin geometry placement. Devoid of lighting, detail, and having only primitive clipping and guidance mechanisms, the first alpha of Frontier was finished and ready for its first full-scale playtest, having until then only been tested by myself.

Again, **please refer to Appendix D** for the top-down layout of Frontier, showing all CP locations (in their final positions.)

3.2: Alpha 1

It was clear within a single round of playtesting that there were serious issues with the altered gamemode. The train was far, far too fast: offensive players steamrolled past the defensive lines with barely a fight. Side routes were given no consideration as the train barreled through. Though the mechanic was found by players to be novel, deep gameplay did not arise and the lack of this inhibited further player interest. Battles between segments were of negligible depth and any further playtest results would have little meaning.

As I returned to the VMF to begin iterating through rapid prototyping, I decided to seek out methods of gameplay data quantification to assist in empirically improving the map, maximizing the positive within the player experience.

Chapter 4

Evaluation Methodology

4.1: Identify Metrics

The first step towards developing such a process requires a review of the metrics available for processing. The Payload gamemode offers a perfect chance to enumerate the progress of a battle: as mentioned before, each payload map contains a track along which are set several control points, to be captured in sequence. As the payload cannot veer from the track, it is a solid metric of offensive progress for every round played on the map. The Source engine's server log functions can be used to track the events each of the 4 CPs fire when capped.

Stop %

The foremost obtainable metric, then, is the number of CPs that have been captured by round end. Tallied from round-to-round and taken over the total number of rounds played on each map, this number becomes the percentage of rounds that end in the stretch of map between any CP and its successor (Stop %). As these percentages are derived directly from playtesting results, they stand as numerical evidence towards the state of the map.

Win %

As the only win condition for BLUE is to capture CP4, with RED winning in any other circumstance, Stop % for CP4 represents BLUE's winning rate. The complement is of course RED's winning rate, with no room for stalemates.

% of Map Experienced on Average

One known variable in each map is the distance of the segments between each CP (see 9.2, *CP Segment Distance* for calculation.) That variable may be used in conjunction with the Stop % of each respective CP to calculate a percentage representing the amount of the total map that is used as a battleground per round on average. A large figure here would mean that the developer effort : user impact ratio is also large; this may however hint at an all-too-easy assault for BLUE and a boring game experience.

Difficulty Peaks

From the captured CP numbers, a unitless figure can be derived representing the respective difficulty of capturing a given CP relative to the other CPs. This is done for the first CP by dividing the percent of games ending before its capture (Stop %) by that figure's complement (Continue %). This produces BLUE's average failure rate (Fail %) for the segment preceding the CP; multiplying this arbitrarily by 100 achieves our final Difficulty rating for it. The second CP's is then calculated by dividing its Stop % by the previous point's complement, and applying the same 100-fold multiplication. The process may be repeated for all 4 points.

Difficulty Valleys

These numbers of arbitrary magnitude represent relative average difficulty for each segment preceding the given CP. It is obvious, however, that within a given segment difficulty is not constant: As BLUE has to fight against the tide of spawning RED defenders, it may safely be assumed that the difficulty of advancing starts out lower at the beginning of every segment, ramping up to a peak at the CP's capture. At that point, spawn locations for BLUE are pushed forward and RED's pushed back, to some degree resetting difficulty—as residual RED players

are killed, that difficulty figure dips down until new waves of defenders arrive. Once that next clash occurs, difficulty would then ramp up once more until the next CP peak. Given the importance of our difficulties' relative magnitudes over their actual number, the low point of a segment can be set arbitrarily at 20% of the average between its end points (CPs). A final valley lies in the humiliation period after a round win.

Player Death Heatmaps

Finally, the Source server logs provide data concerning every player death that occurs on a map. This data may be plugged through a third-party tool to generate a 'heatmap' of deaths over a top-down overview of the map itself. Darker areas correspond to regions of high battle intensity, with lighter regions seeing less action. Such heatmaps shed light on intensity changes in a 2D format, allowing for less-precise but easily formulated conclusions about the map's gameplay.

4.2: Identify Variables

With definite metrics decided, determining equally quantifiable independent variables becomes the next step towards iterating empirically.

CP Segment Distance

The first variable measurable is the physical distance between each CP. By setting the speed of the payload manually, the time required to travel a CP can be derived as the payload reaches it. Within Hammer, 1 unit is equivalent to 1 real world inch; speed in Source is in the form Hammer Unit over Time in Seconds. Therefore, at a speed of 1 the payload will deliver track segment distances directly in Hammer Units (or inches.) Having measured the segments

between each CP, the resultant numbers can then be taken one-by-one over their total sum to achieve a percentage representing a segment's length relative to the total track length. In combination with payload speed, segment length is a figure contributing to that segment's difficulty.

Payload Train Speed

Set directly in Hammer, the payload train's speed is easily known. As its speed determines how fast offense may move forward, that speed holds implications for defense's ability to recoup, and the map's difficulty as a whole. Payloads increase speed on 4 intervals: with 0 players pushing, with 1, with 2, and with 3 or more. Adjusting each of these could significantly alter the way the map plays out.

Spawnwave time

In TF2, dead players are spawned in 'waves' to encourage rolling out of a spawn location in a more team-like fashion. Based on a variable specific to each team, the spawnwave time, the server will wait X seconds before attempting to spawn all currently dead players. As players can die anywhere within this interval, it is safe to say that a recently dead individual will on average be resurrected after half of their team's spawnwave time has passed. Adjusting this variable for either team determines the 'clashing point' for an area. A clashing point is the spot at which two opponents will meet on the map if they die at the same time. As clashing points generally define the area around which the battle's front line shifts, small changes in spawnwave time can cause ripples of gameplay impact.

Layout Alterations

Of course, there are nearly infinite unquantifiable changes that could be made within the map. Geometry itself can provide combat advantage or disadvantage depending on layout decisions, but a numerical representation of geometry changes cannot be cleanly derived. That said, there are time-tested ways of giving advantage to offense or defense selectively:

1. Gating: Using one-way passageways to allow the bypassing of high-traffic areas without fear of repercussion. These are most often seen in the form of actual one-way gates to offense's advantage, and dropdowns to defense's.
2. Dropdowns: An instance of gating simply using a hole as a pathway from a higher area to a lower one, dropdowns provide opportunity for surprise attack.
3. Height advantage: Due to splash damage's effectiveness on ground surfaces, height advantage can be a huge combat bonus—especially in TF2. Gating into higher spaces from a particular direction can direct this advantage towards the team approaching from that angle.
4. Cover: Providing geometry to hide behind helps a team recoup in a given region. It can also absorb incoming fire, or obscure a long sightline being abused by long-range weaponry.
5. Windows: In a way, a window represents a form of inverse cover, providing safety around its edges in reference to activity passing through its open space. As a result, players shooting through a window from close proximity have a stronger combat position than those further away. This fact can be used to create areas of defensive strength, or high-value offensive bombardment locations.
6. Pickups: The movement of health and ammo can define areas of safety or create points around which teammates meet up before resuming battle. Placing

these towards either team's respective side of the battlefield can confer a supply-line advantage, keeping players "fresh" for the fight ahead.

4.3: Identify Goals

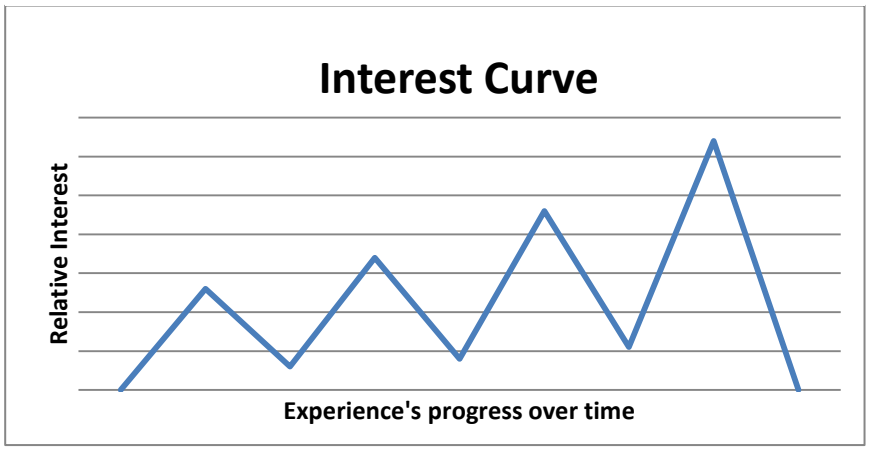
Having both variables to adjust and metrics by which to judge the results leaves but one thing to be decided: goals to steer those results towards. Precedent and literature both play a role here.

Interest Curves

In any entertainment experience, the intensity of the emotion at hand—horror in a haunted house, excitement on a roller coaster, fun in a game—can be graphed as a function of time. This produces a curve representing the engagement level of the audience with the experience itself. It stands as evidence to the effectiveness of the experience in producing that emotion (Schell, 2008/2010, pp. 245-252).

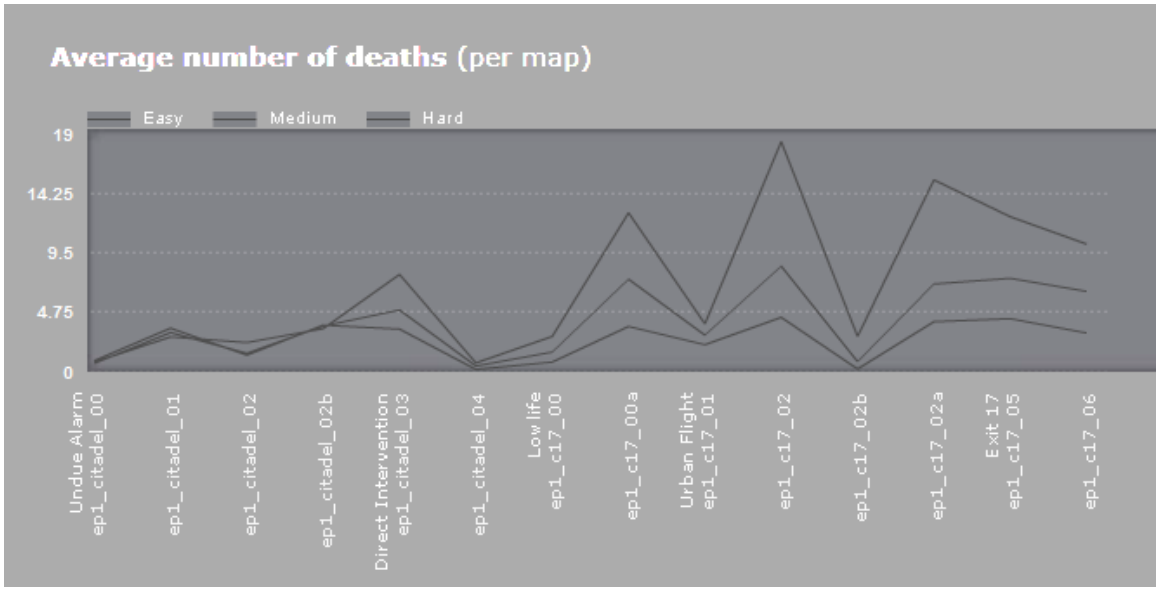
The most successful interest curve is neither a straight line, nor a bell curve—it is a series of gradually higher peaks, with valleys of lower intensity in between each pair of high points. An audience—in the case of a game, its players—are quick to be overloaded by interest if not given periodic breaks between peaks. Without these breaks, peaks cannot develop at all, resulting in the loss of the "epic" feeling created by a gameplay experience that gradually peaks higher and higher until its conclusion and resolution (Schell, pp. 245-252). An ideal (if arbitrarily so) interest curve might resemble the following:

Figure 4.3.1



As there exists no unit to judge 'interest,' this variable can vary from media to media. Speed would be a good predictor of interest on a roller coaster. In a video game that interest factor is most often associated directly with the game's difficulty at a given point. How that difficulty is measured can differ itself; in this example, average player deaths stands as evidence towards difficulty per map in Valve's *Half-Life 2: Episode 1*:

Figure 4.3.2 © 2011 Valve Corporation. Used with permission.



While perhaps not ideal, this is a solid interest curve demonstrating the game's progressively harder conflicts, separated by interludes of story or equivalently easy gameplay (Schell, pp. 245-252).

Map Experience

Within the context of Frontier, the interest curve can be represented by the aforementioned peaks and valleys of difficulties surrounding the capture and successive battle lulls over each CP in turn. Having quantified these previously, it is a simple matter to then convert them into "difficulty curves" demonstrating the map's difficulty and therefore interest value over its course, relative to the attacking BLU team.

As RED's difficulty curve is the complement of BLU's, it would initially seem that the RED experience is opposite of the ideal. However, BLU's line can also be said to represent the RED experience in a sense. On defense, the only objective that matters is that the last CP not be captured before time runs out—all preceding CPs are simply opportunities for defense to stop the assault. To BLU, however, each point is as important as the next, as each must be captured for the team to win. Because of this difference, there is a pressure coefficient augmenting the difficulty value of each CP's defense. While an easily captured CP1 might seem too hard for RED to defend successfully, with little pressure to hold at 1 the experience is made less intense. As this coefficient approaches 1 in relationship to map progress, it is suitable to use BLU's difficulty curve as the interest curve for all players in the game.

The first definite goal, then, is to alter the map in such a way that its difficulty curve approaches the ideal interest curve form. As the ideal may differ from author to author depending on intent, its construction may be relatively arbitrary—that is, should it follow the general form for a well-crafted experience of rising peaks and valleys.

The second goal relates to fairness—even in assault-oriented levels, fairness can be hard to quantify. As one team ultimately has to win and one to lose, defining a point at which the map may be considered fair to each role can be difficult. In these cases, it helps to tier level goals—

that is, to provide multiple goals that must be accomplished sequentially in order to progress to the final goal. This can give developers a better metric for determining assault effectiveness, as well as periodically rewarding (emotionally) the offensive players who are otherwise fated to lose. Of course the payload format assists towards this, but in the end there must always be a winning rate for each team. In pursuing the goal of fairness, pushing the average wins for the map towards an even split becomes an objective.

The third and final design goal in this case study refers to how much of that interest curve is experienced per map play—the aforementioned *% of Map Experienced on Average* figure. Maximizing this without sacrificing interest curve integrity nor winning rate is the final criterion on which iterations will be judged.

4.4: Control

The aforementioned Valve map Badwater provides a perfect parallel to Frontier; its similarities in CP number, track length and generally open layout make it an excellent control during Frontier's development process. The values provided for Badwater's metrics offer context for those generated by each new Frontier revision. Reaching or surpassing the standards set by Badwater in each examined result area stands as a solid stop goal for development.

The first number on the next page represents the average percentage of every round played that ends with a victory for BLU/offense. The next is the calculated percentage of the entire map that is experienced—in other words, in which gameplay takes place—on average per round. The third is the degree in percentage to which the difficulty (or interest curve) ideals from the stated "ideal" curve.

The first figure shows the deaths of all players in the sampled space of time displayed in heatmap form over the map's overview. "Hotter" areas represent areas of more frequent player

death, and therefore hint towards activity on the map as a whole. The second figure displays the % of games that ended on the map in respect to each Control Point being the last captured beforehand. The third graph shows the respective difficulty of each CP distilled from the *Stop %*. The final graphs those difficulties over the course of the map, creating a Difficulty Curve that maps to the level's Interest Curve.

This template will also be used in the later breakdown of the case study's iteration results.

Badwater Summary:

39% Offensive Wins

83% of Map Experienced on Average

13% Deviation from Ideal Curve

Figure 4.4.1: Player Death Heatmap

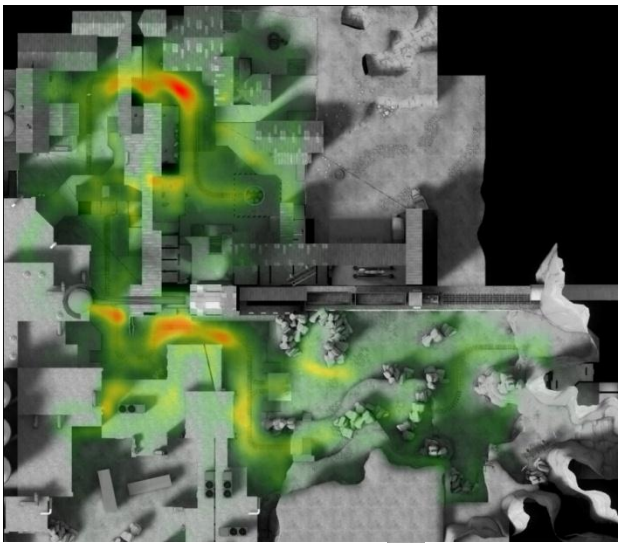


Figure 4.4.2

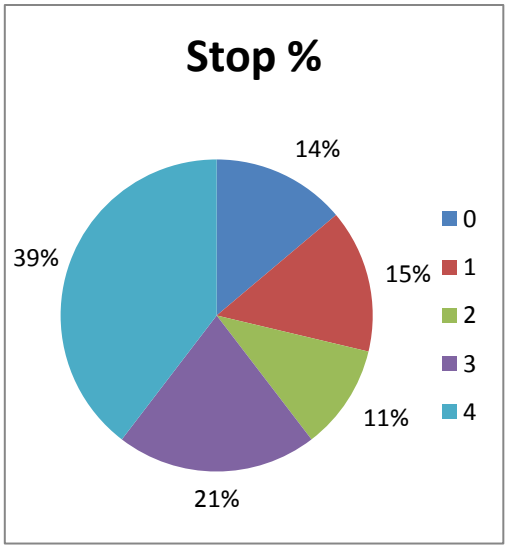


Figure 4.4.3

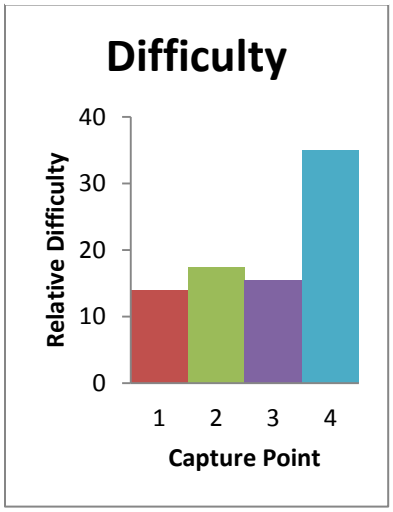
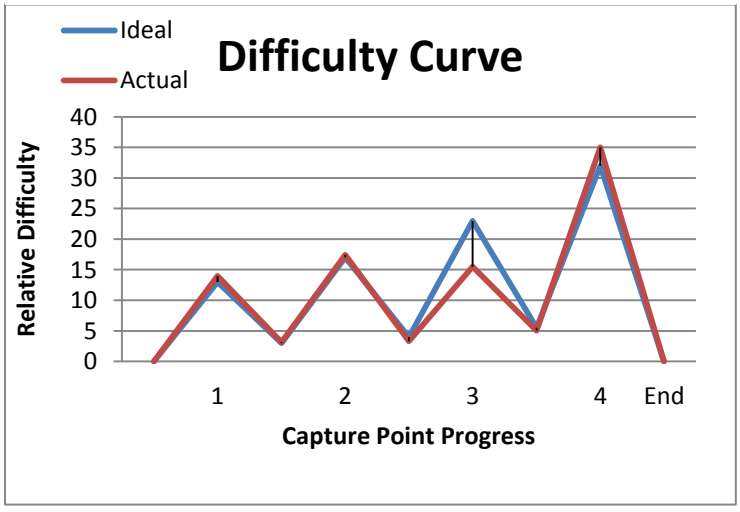


Figure 4.5.4



4.5: Identify Hypotheses

If playtesting is viewed as an experimental process, our game variables become the independent variables and our metrics the resulting dependent variables. Through this lens, the changelogs between each map iteration represents the experiment's hypothesis.

It should be noted that as multiple level design techniques can and should be used to increase a map's playability, the effects of each changed variable may be synergistic or even confounding with one another. This can make the partitioning of each into strict cause-and-effect relationships very difficult. Therein lies the 'art' of level design theory—skillfully predicting exactly how a full changelog of adjustments will manifest itself in gameplay terms.

With time allowed in the development cycle for testing the many combinations of said adjustments, rapid prototyping can help take the guesswork out of the relationship. This process consists of quickly issuing changes, collecting playtest data, and then analyzing it and repeating. So while the variables may not be inseparable, measurable results will still represent the sum of their impact.

It is important to note that these "hypotheses" are useful only in the context of a single iteration / playtest cycle; an example set of hypotheses might read as the following:

H_1: Lowering spawn times for RED during Segment 2 will lead to a higher *Stop %* at that point.

H_2: Increasing the length of Segment 3 relative to the other segments will lead to a higher % of the map being played on average.

This is the mindset with which every changelog in the following chapters should be viewed: as a series of hypothesized "improvements" in the targeted areas as a result of geometry and other gameplay changes.

Chapter 5

Rapid Prototyping Decisions & Results

This chapter will cover, in-depth, every iteration of the map occurring within the rapid prototyping phase of development. This spans from the second version of the map up to the one preceding the first major artpass. After an artpass, the exponentially more complex geometry of the map makes large changes to layout much harder to implement on the developer's end; thus, the majority of layout alterations will be found within this chapter.

For each iteration, statistics will be provided for all measured variables. A changelog detailing layout and geometry decisions will accompany these. The results expressed in our chosen metrics follow, with brief analysis of their significance.

Testing Environment

While the default server size for TF2 is 24 players, the larger 32-player format has significant popularity. The extra players, however, can cause crowding within maps not designed towards this demographic, resulting in spammy unskilled gameplay. Accounting for this in Frontier would be necessary.

A single 32-player server was chosen for the empirical portion of playtesting efforts. This particular server had sufficient popularity to guarantee full playtests at nearly any time. The server was devoid of any gameplay-altering plugins that might impact results. Although there were never more than 32 players playing the map at once, the total number of unique players has been listed per iteration in the respective Results Summary sections.

Please reference **Appendix B** for the equations used on the data in **Appendix C** in generating the following graphs and figures.

5.1: Iteration 1

Version: pl_frontier_a2 (Alpha 2)

Undetailed prototype

Changelog:

- Removed central RED spawn in favor of a forward spawn next to CP1.
- Second RED spawn moved into a separate room; former space used as side route, with a dropdown into the space before CP2.
- Added another exit from first BLUE spawn along with space to its right to avoid the congestion in this area in A1.
- One-way path added from bottom of warehouse after CP1 to area behind CP2.
- One-way path added from cliff overlooking CP1 to left side to hill preceding CP2.
- Shortcut added for RED when spawning from CP4; door closes after CP2 is captured.
- Train speed greatly reduced.
- Spawnwave times manipulated.

Figure 5.1.1

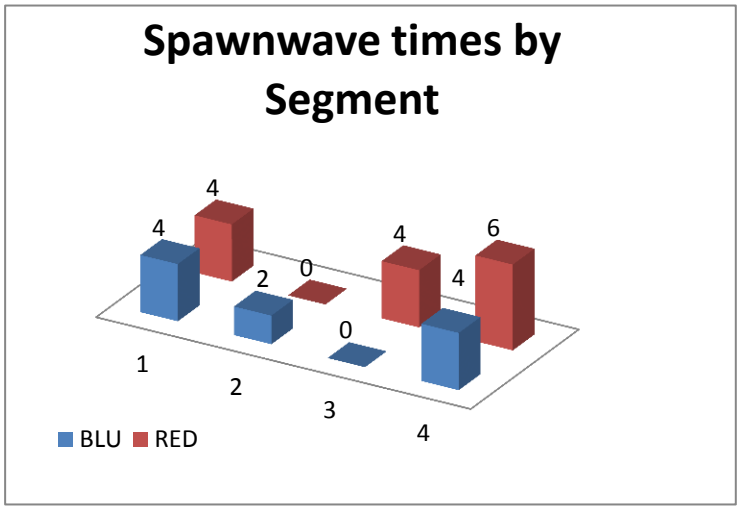


Figure 5.1.2

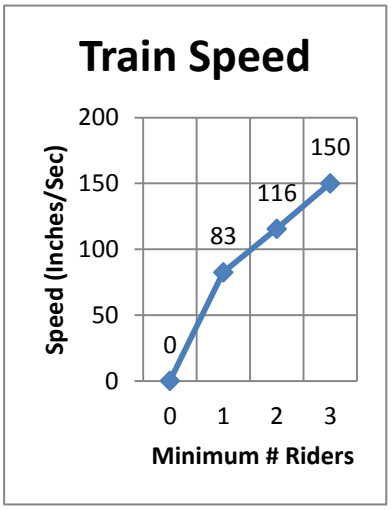
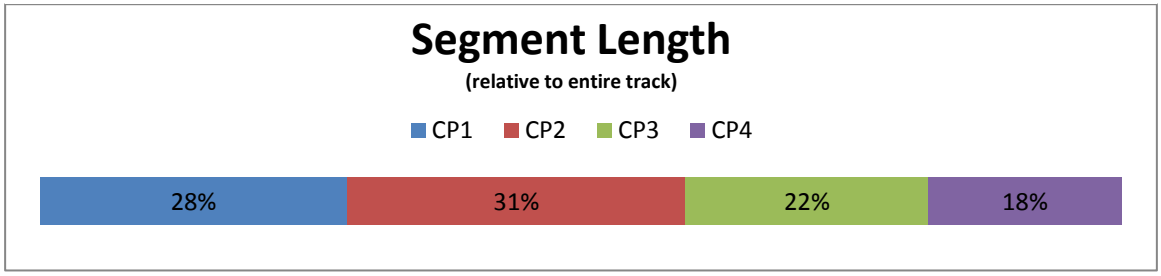


Figure 5.1.3



Iteration 1 Playtest Results Summary:

46 Unique Playtesters

39% Offensive Wins

54% of Map Experienced on Average

205% Deviation from Ideal Curve

Figure 5.1.4: Player Death Heatmap

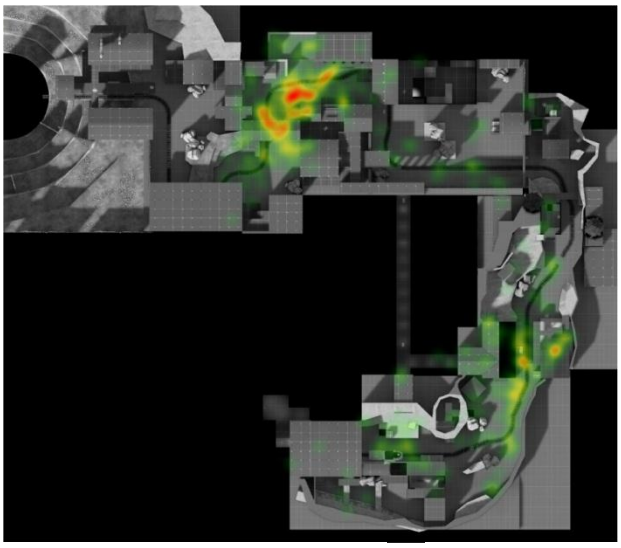


Figure 5.1.5

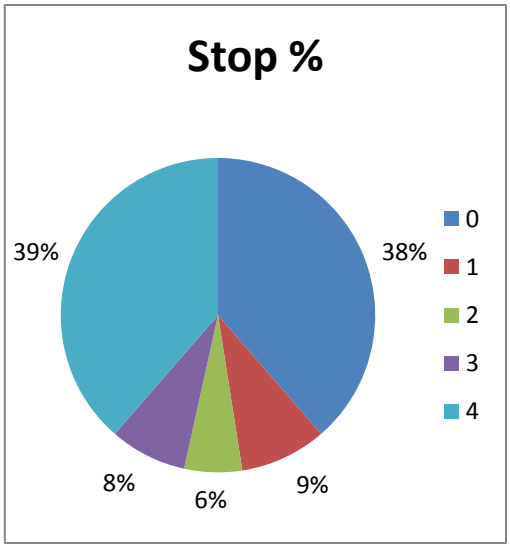


Figure 5.1.6

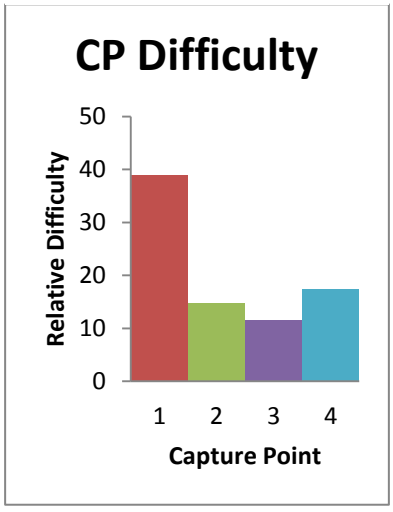
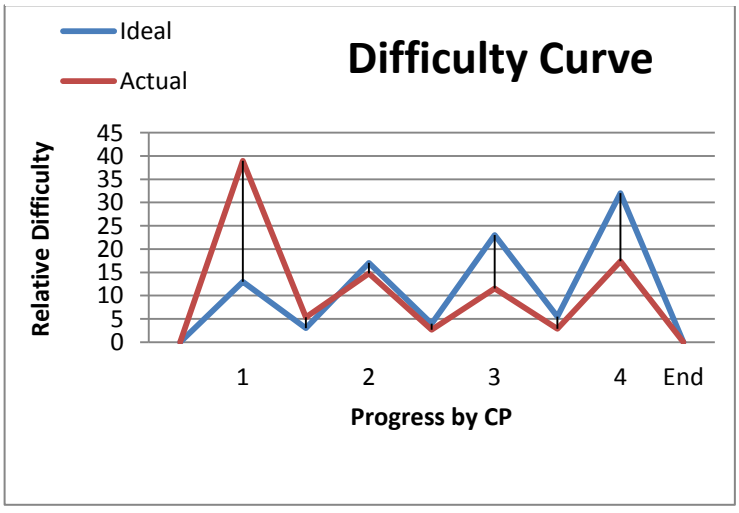


Figure 5.1.7



5.2: Iteration 2

Version: pl_frontier_a3 (Alpha 3)

Undetailed prototype

Changelog:

- First RED spawn set to become a BLU spawn 10 seconds after CP1 is captured.
- BLU shortcut from first spawn to CP2 area after capturing it removed in favor of a new spawn location at shortcut's former exit.
- Spawnwave times manipulated in defense's favor.

Figure 5.2.1

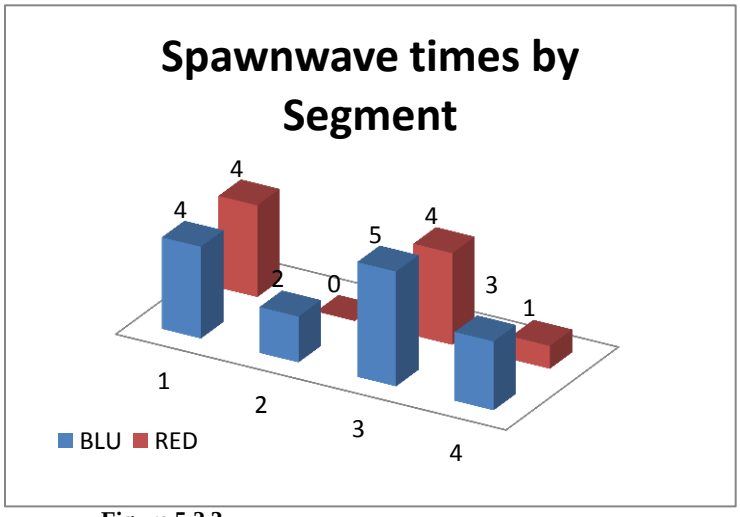


Figure 5.2.2

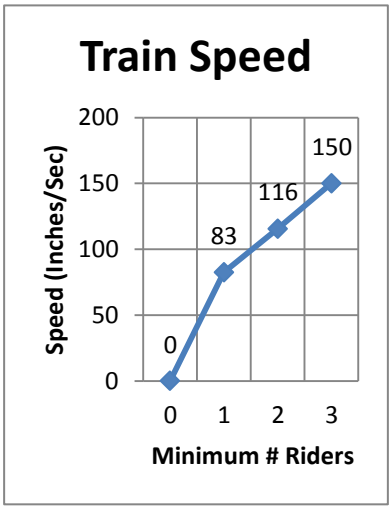
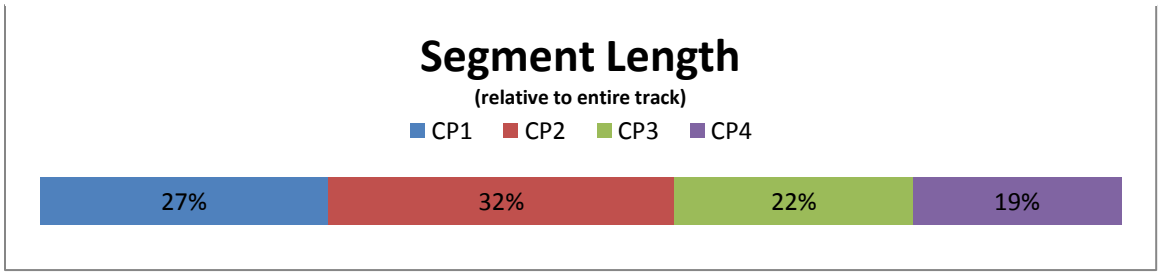


Figure 5.2.3



Iteration 2 Playtest Results Summary:

133 Unique Playtesters

42% Offensive Wins

75% of Map Experienced on Average

71% Deviation from Ideal Curve

Figure 5.2.4: Player Death Heatmap

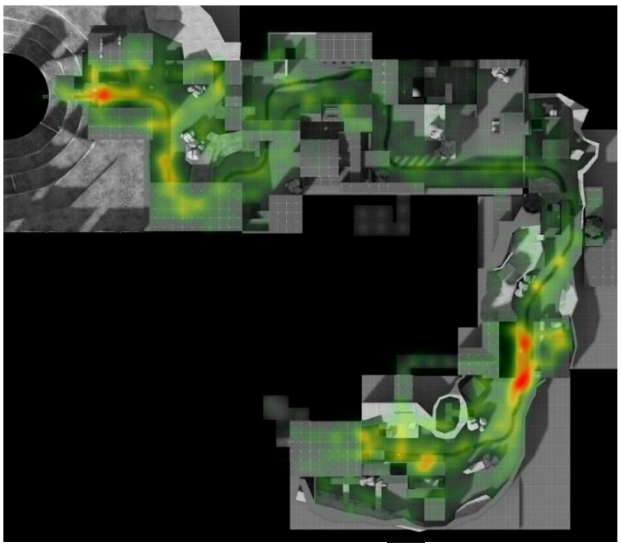


Figure 5.2.5

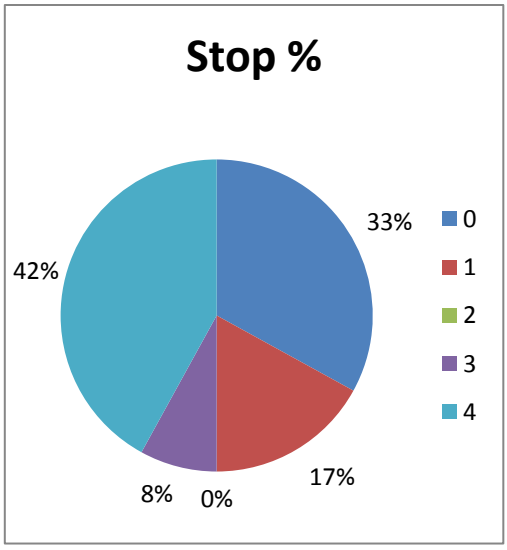


Figure 5.2.6

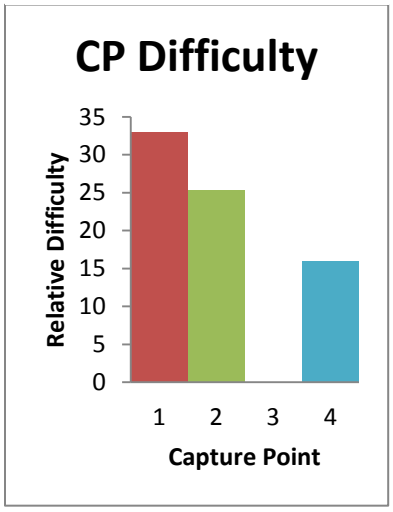
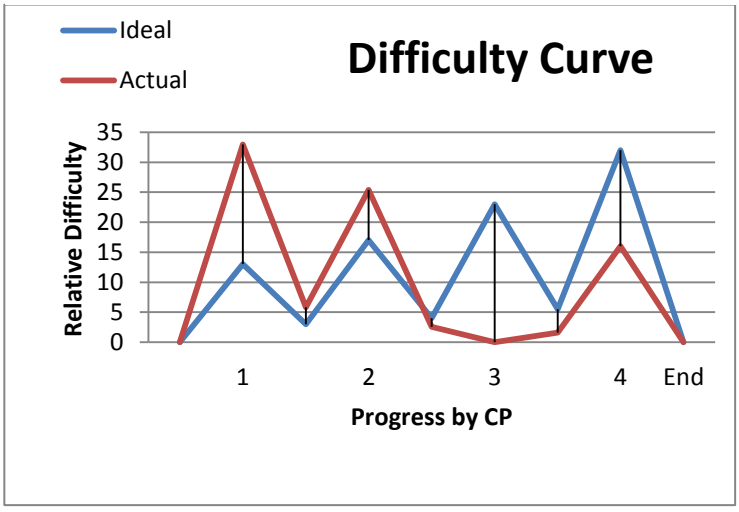


Figure 5.2.7



5.3: Iteration 3

Version: pl_frontier_a4 (Alpha 4)

Undetailed prototype

Changelog:

- Added left exit from BLU first spawn past first segment of track.
- Ramp added leading into building preceding CP1 choke.
- Tunnel dug through cliff on right of aforementioned choke to relieve congestion.
- Added path from cliff above CP1 connecting to top exit of second BLU spawn.
- Restructured stairs leading from warehouse around CP2.
- Platforming rearranged below final CP for BLU's benefit.
- Path from dropdown before CP2 opened out onto overlooking balcony.
- Spawnwave times generally increased.
- Train speed again cut.

Figure 5.3.1

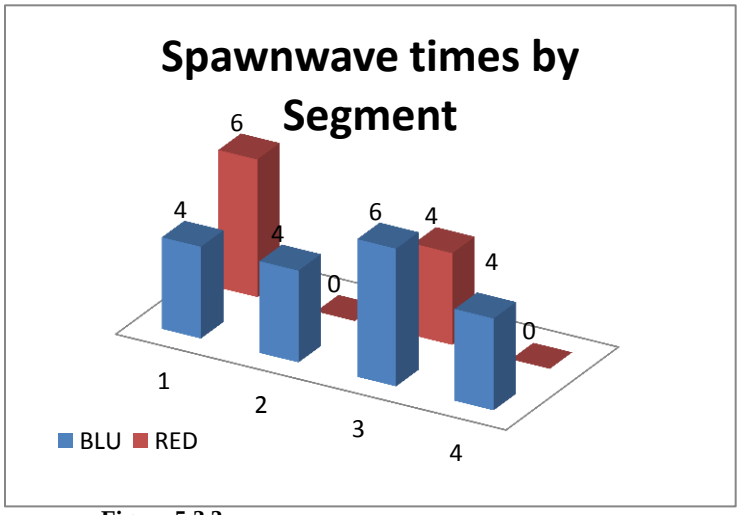


Figure 5.3.2

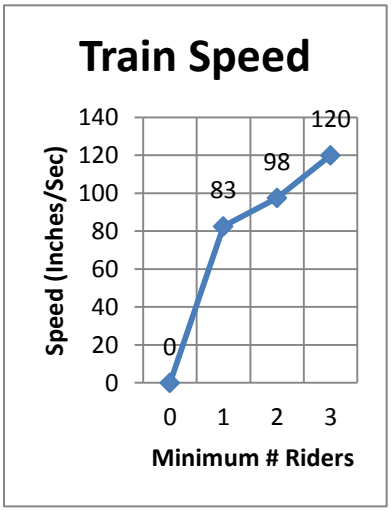
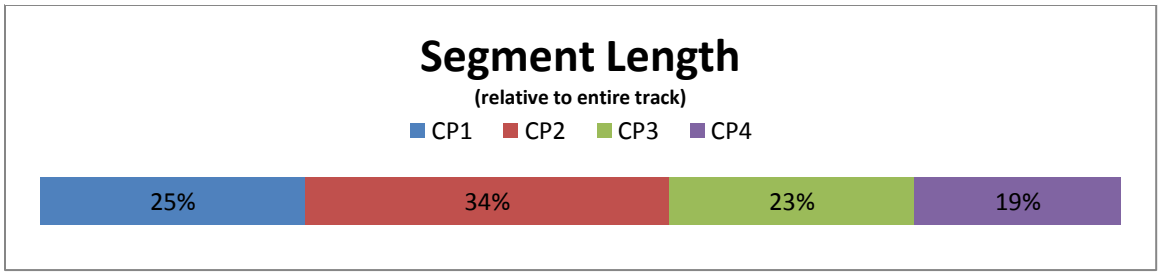


Figure 5.3.3



Iteration 3 Playtest Results Summary:

154 Unique Playtesters

20% Offensive Wins

64% of Map Experienced on Average

106% Deviation from Ideal Curve

Figure 5.3.4: Player Death Heatmap

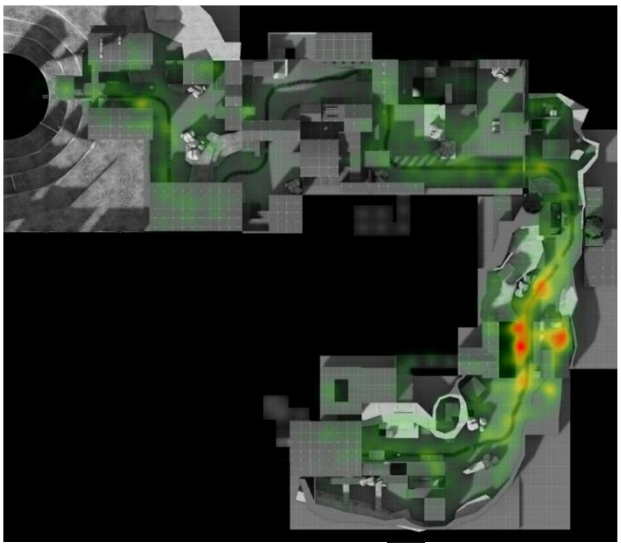


Figure 5.3.5

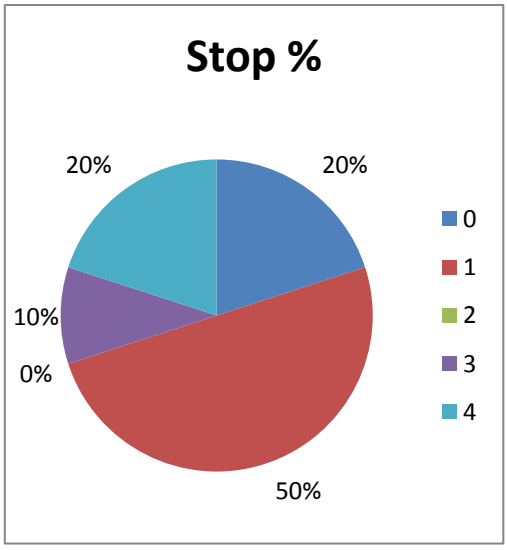


Figure 5.3.6

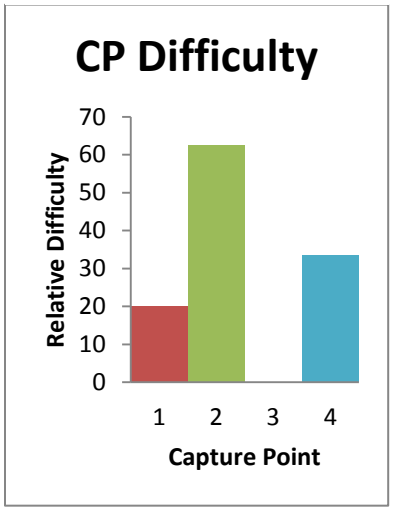
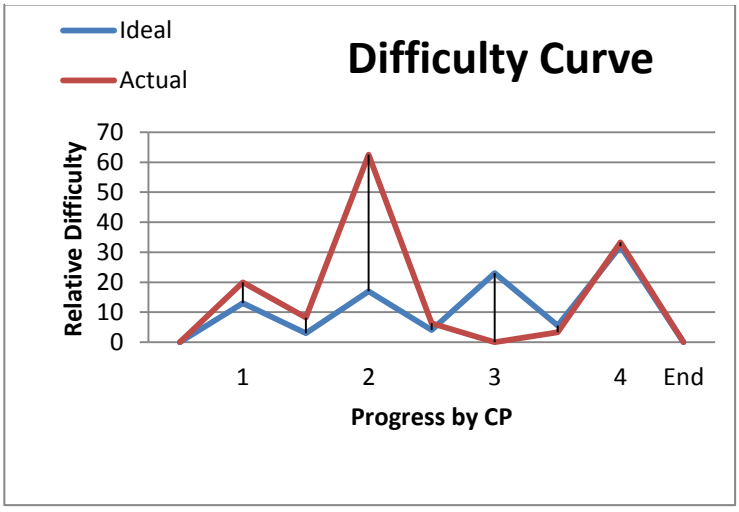


Figure 5.3.7



5.4: Iteration 4

Version: pl_frontier_a5 (Alpha 5)

Undetailed prototype

Changelog:

- Window added to building before CP1 allowing offense to pick off enemy sentries.
- Rightmost pathway onto hill after CP1's left exit sealed off to provide cover and conceal players taking that route.
- Spawnwave times adjusted in favor of BLU.

Figure 5.4.1

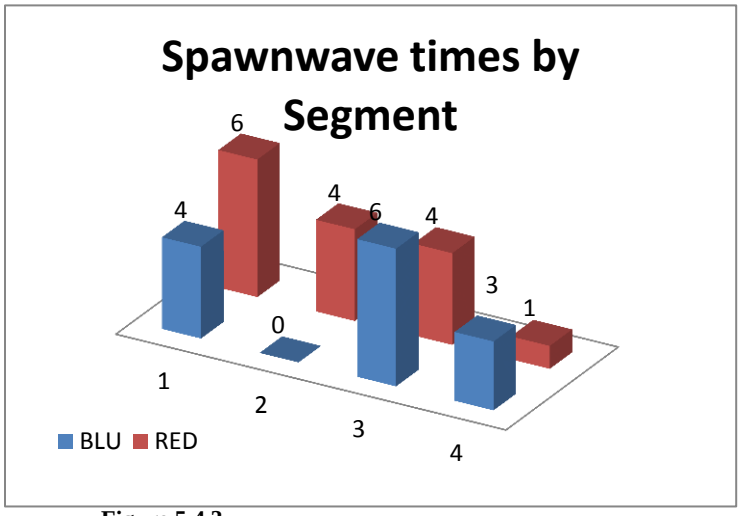


Figure 5.4.2

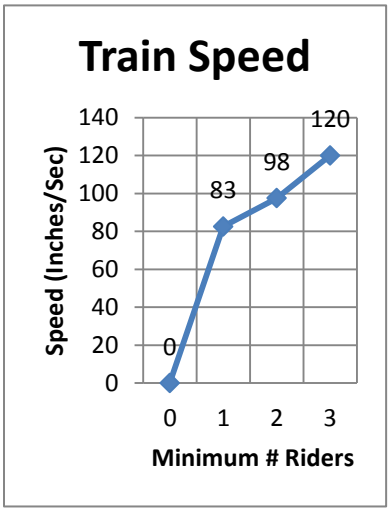
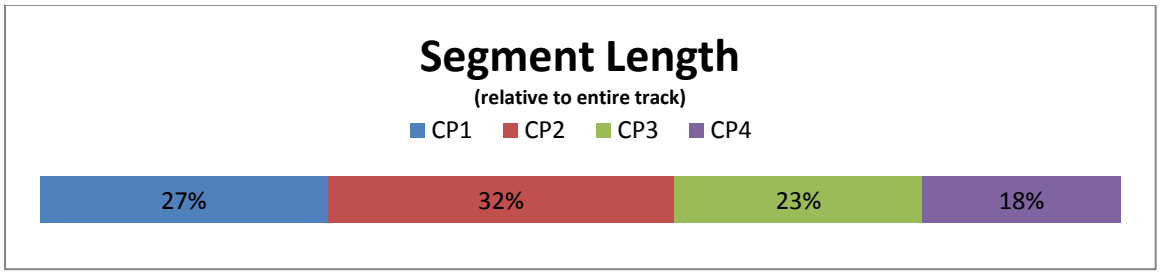


Figure 5.4.3



Iteration 4 Playtest Results Summary:

141 Unique Playtesters

25% Offensive Wins

68% of Map Experienced on Average

67% Deviation from Ideal Curve

Figure 5.4.4: Player Death Heatmap

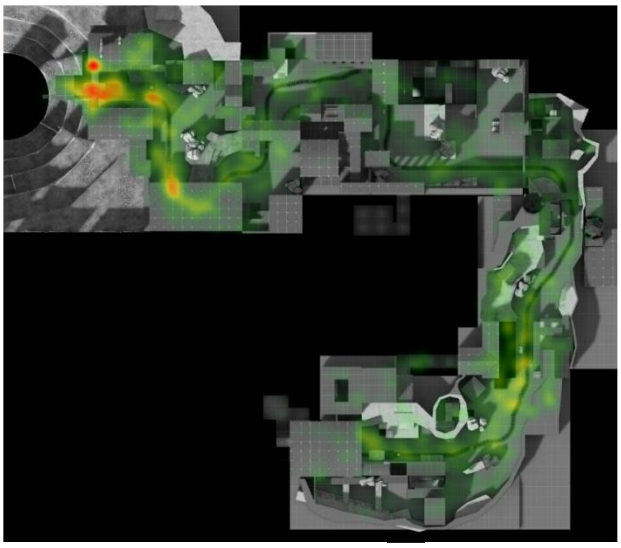


Figure 5.4.5

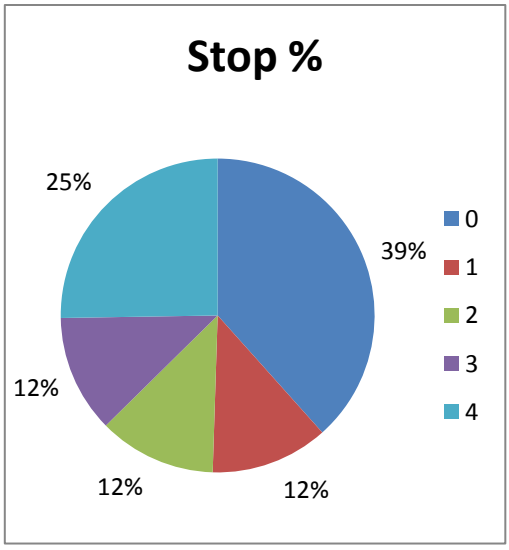


Figure 5.4.6

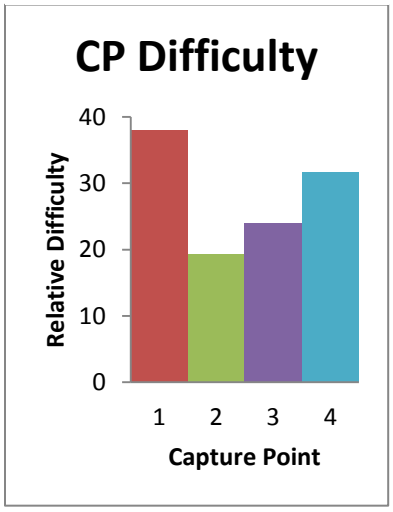
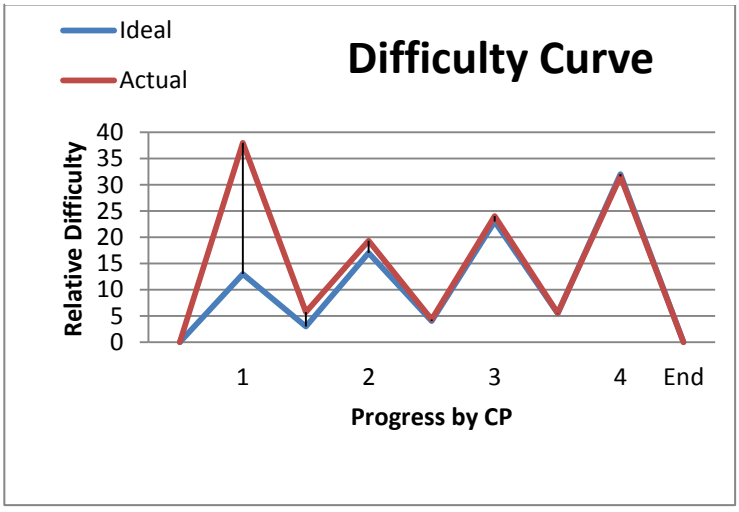


Figure 5.4.7



5.5: Iteration 5

Version: pl_frontier_a6 (Alpha 6)

Contextually detailed prototype

Changelog:

- Contextual detailing added temporarily.
- Bridge added from cliff overlooking CP1 to path leading to rightmost entrance onto hill preceding CP2.
- CP2 warehouse bottom floor pathway moved up to top level.
- Path leading from space below CP4 around and back up to groundlevel.
- Ramps leading to top floor of CP3 warehouse added to its left side, allowing offense to gain higher ground inside.
- Entrance to former RED spawn between CP2 and CP3 removed to make holing up within easier.
- Roof added directly above CP4.
- CP4 moved back by roughly 25%.
- Spawnwave times manipulated, with RED's becoming generally longer.

Figure 5.5.1

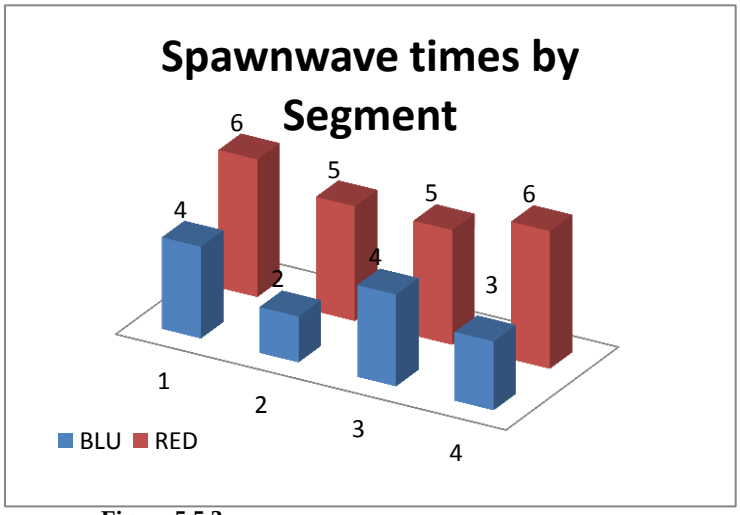


Figure 5.5.2

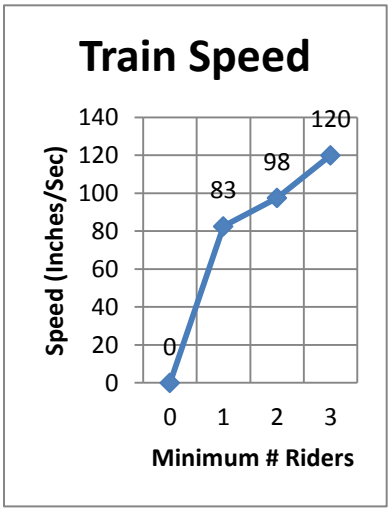
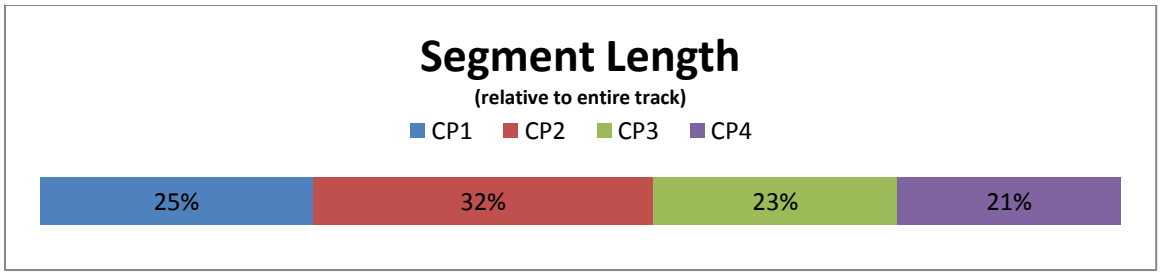


Figure 5.5.3



Iteration 5 Playtest Results Summary:

57 Unique Playtesters

50% Offensive Wins

84% of Map Experienced on Average

73% Deviation from Ideal Curve

Figure 5.5.4: Player Death Heatmap

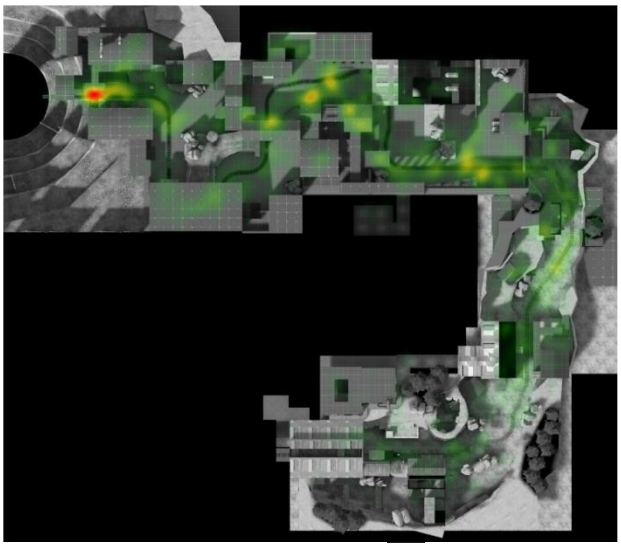


Figure 5.5.5

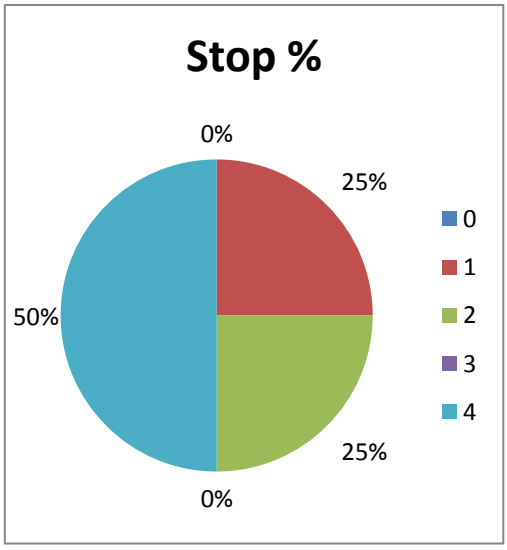


Figure 5.5.6

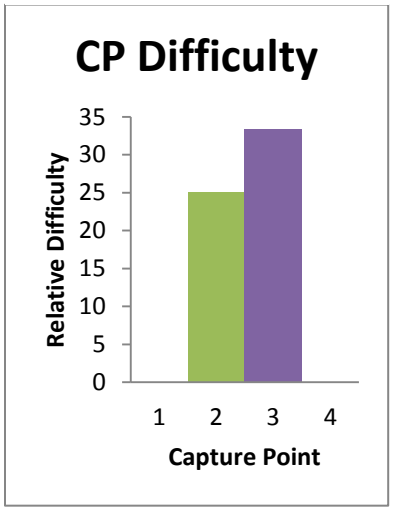
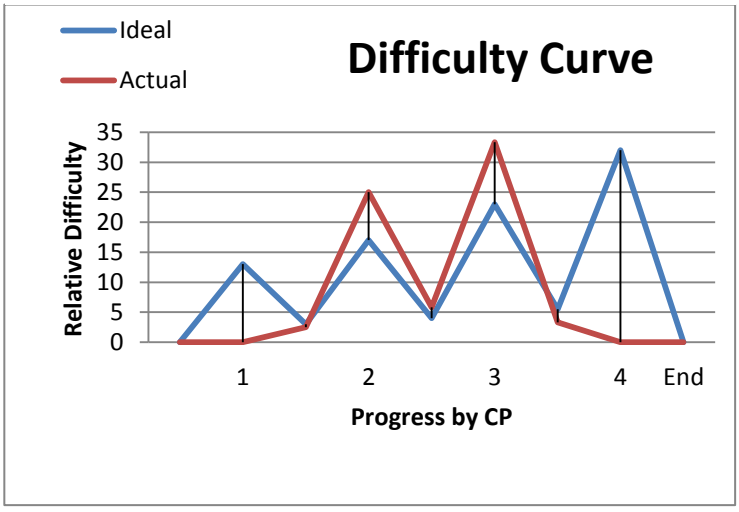


Figure 5.5.7



5.6: Iteration 6

Version: pl_frontier_a7 (Alpha 7)

Undetailed prototype

Changelog:

- No significant changes to geometry made.
- Clipping mistakes corrected.
- CP4 moved forward again slightly.

Figure 5.6.1

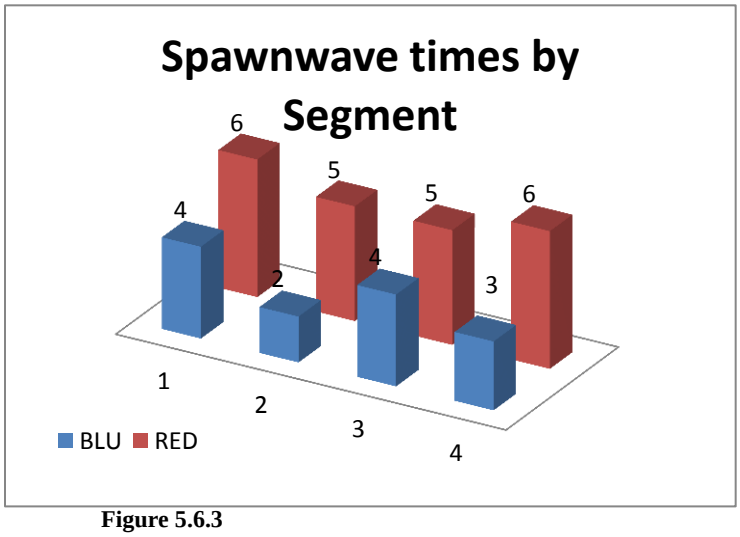


Figure 5.6.2

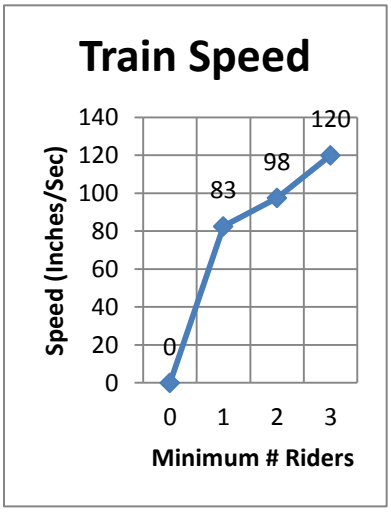
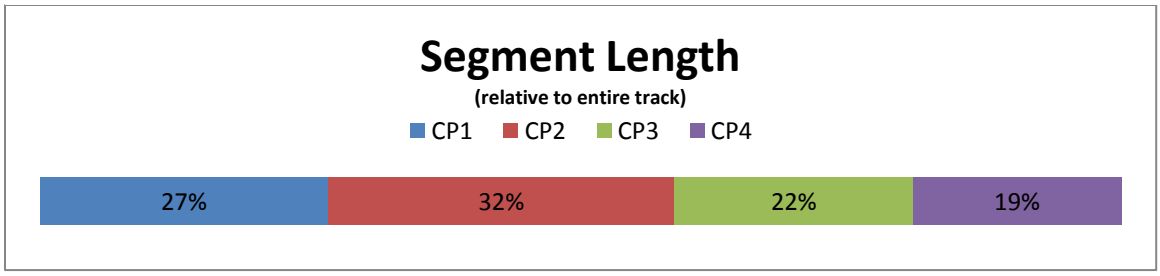


Figure 5.6.3



Iteration 6 Playtest Results Summary:

71 Unique Playtesters

25% Offensive Wins

77% of Map Experienced on Average

73% Deviation from Ideal Curve

Figure 5.6.4: Player Death Heatmap

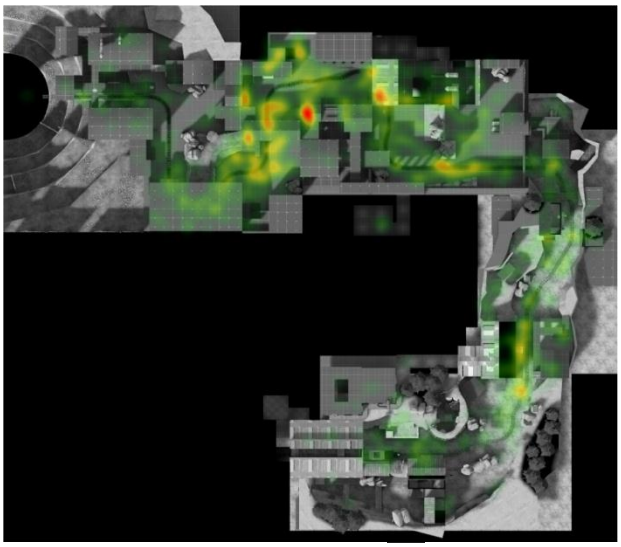


Figure 5.6.5

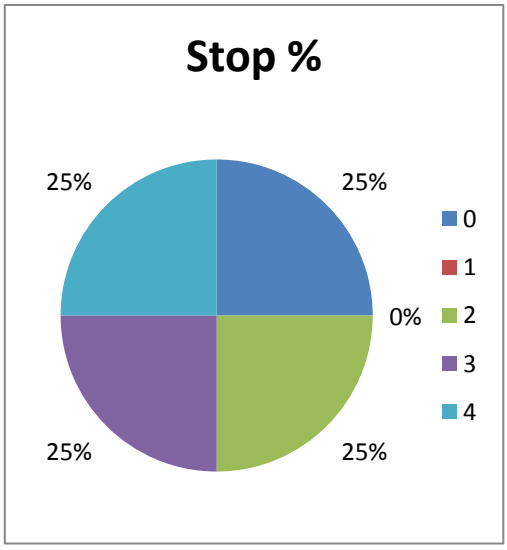


Figure 5.6.6

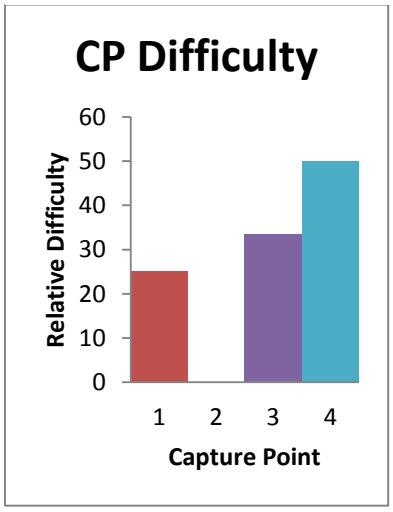
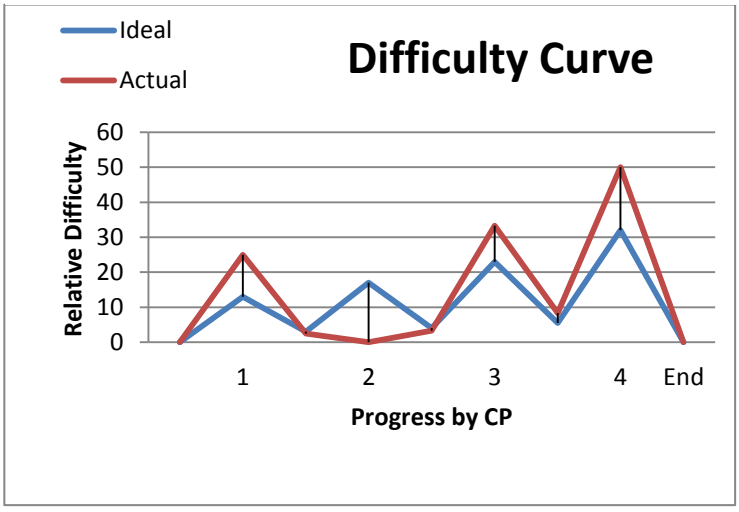


Figure 5.6.7



5.7: Iteration 7

Version: pl_frontier_a8 (Alpha 8)

Contextually detailed prototype

Changelog:

- Doors added to block off path from CP2 warehouse past CP2 until it is captured.

Figure 5.7.1

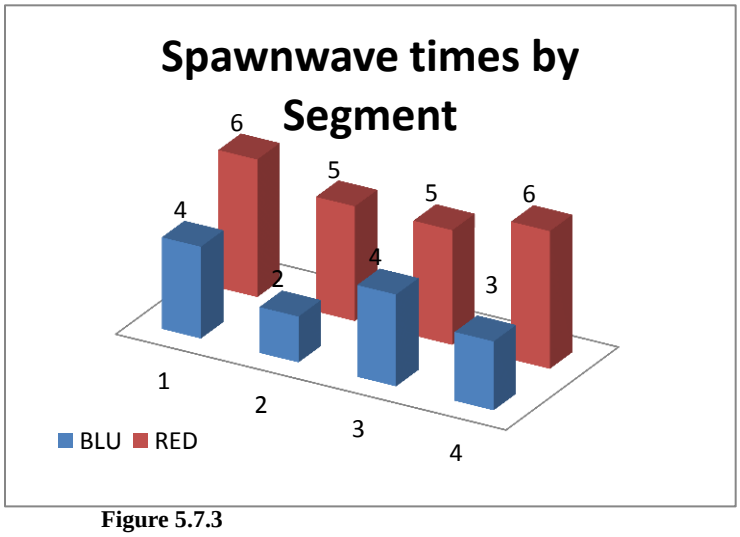


Figure 5.7.2

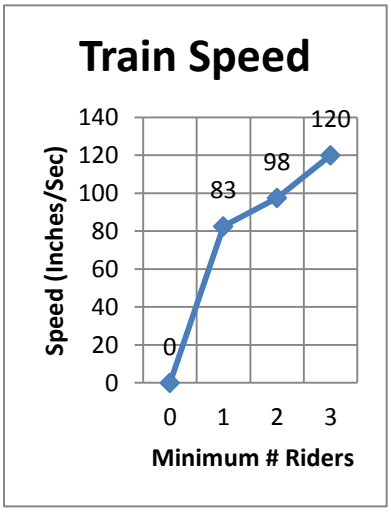
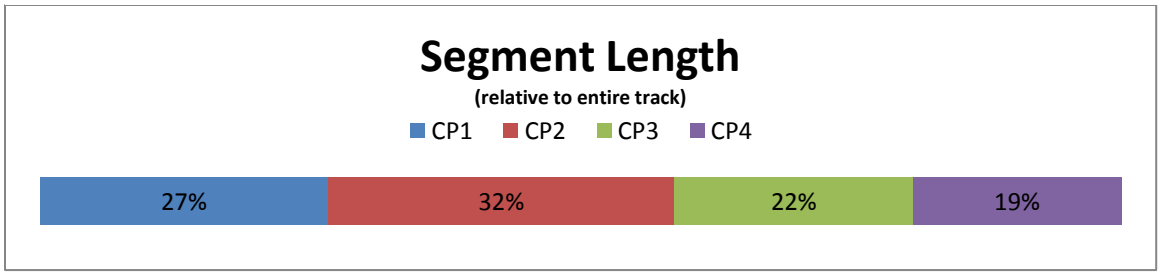


Figure 5.7.3



Iteration 7 Playtest Results Summary:

213 Unique Playtesters

59% Offensive Wins

73% of Map Experienced on Average

107% Deviation from Ideal Curve

Figure 5.7.4: Player Death Heatmap

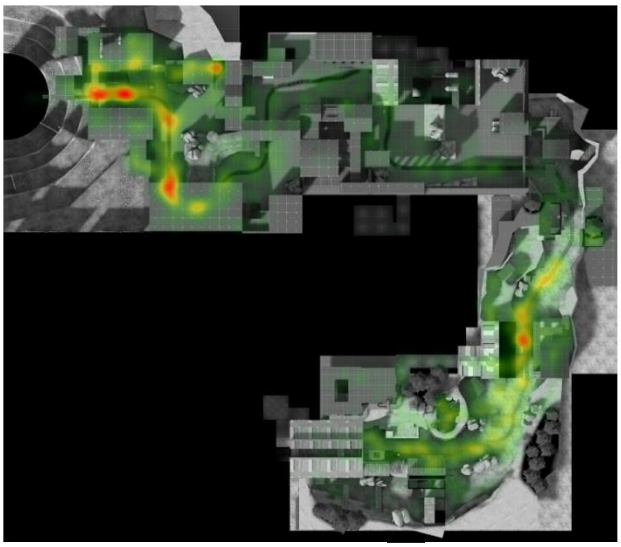


Figure 5.7.5

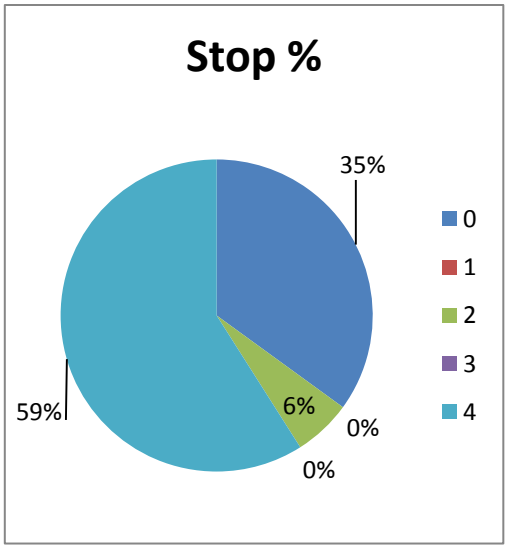


Figure 5.7.6

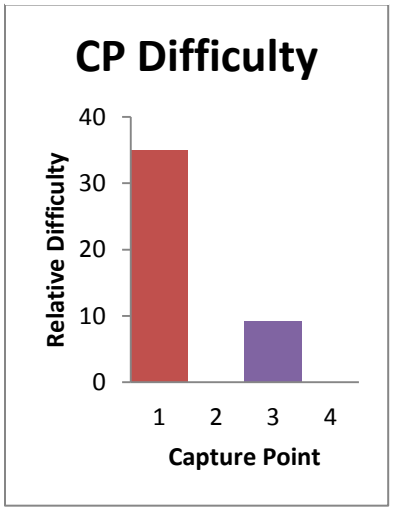
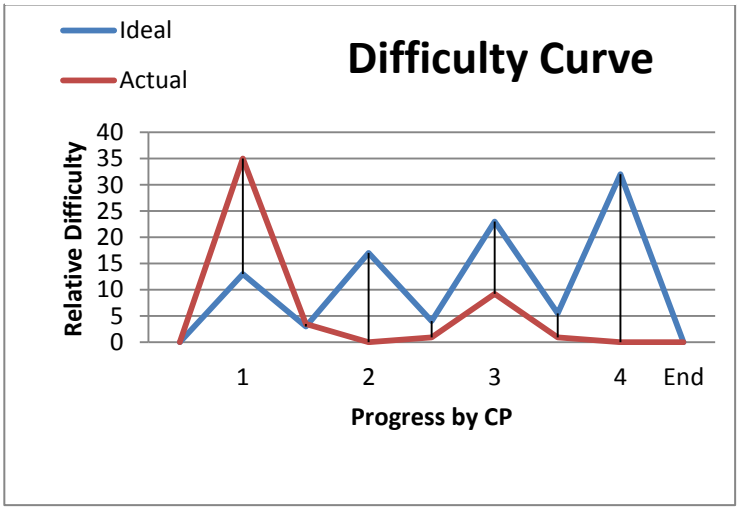


Figure 5.7.7



5.8: Iteration 8

Version: pl_frontier_a9 (Alpha 9)

Contextually detailed prototype

Changelog:

- Path from leftmost exit of BLU first spawn to building next to segment 1 choke diverted into earlier rock structure. Stairs added into this building from the ground floor.
- Second gallery window added below first within aforementioned building.
- Cliff pathway above CP1 diverted into nearby building.
- Windows cut into bridge structure before segment 2 hill.
- Segment 2 hill right entrance expanded into two-story building with dropdown into entrance.
- Ramp from defense's side of choking tunnelway in segment 3 added dropping down into former red spawn.
- Ramp from defense's side of tiered rock structure added up to top level; sliding door shortcut removed in compensation.
- Hole cut into roofing above CP4 for dropdown access.

Figure 5.8.1

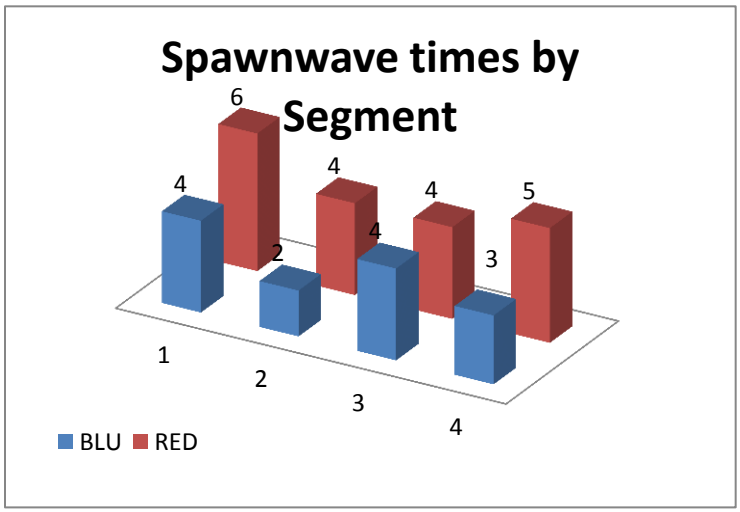


Figure 5.8.2

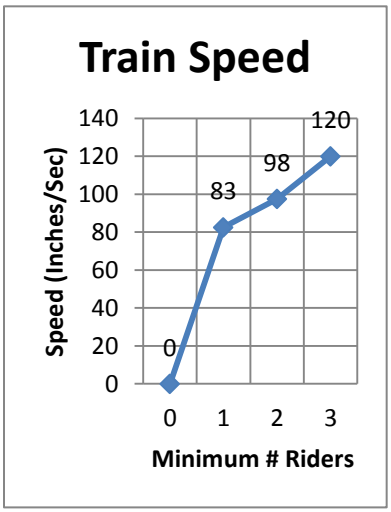
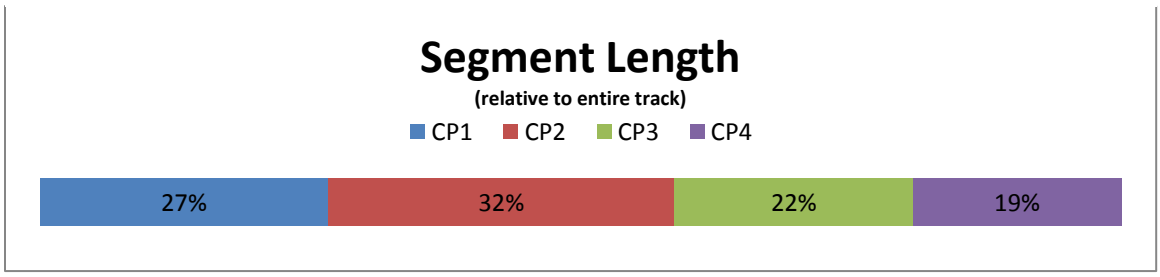


Figure 5.8.3



Iteration 8 Playtest Results Summary:

111 Unique Playtesters

66% Offensive Wins

90% of Map Experienced on Average

67% Deviation from Ideal Curve

Figure 5.8.4: Player Death Heatmap

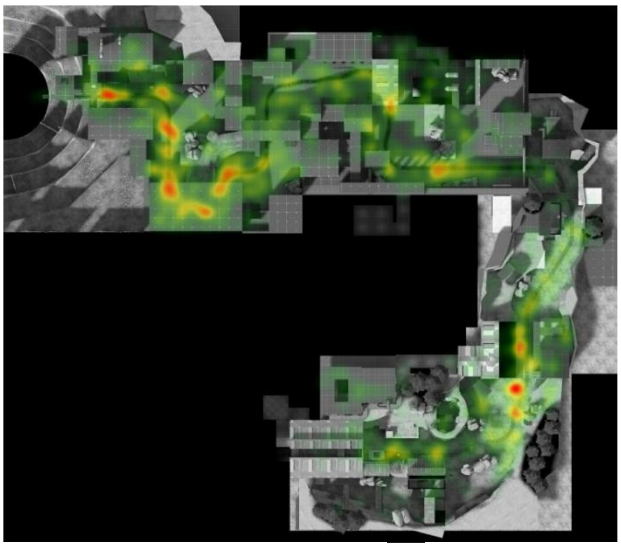


Figure 5.8.5

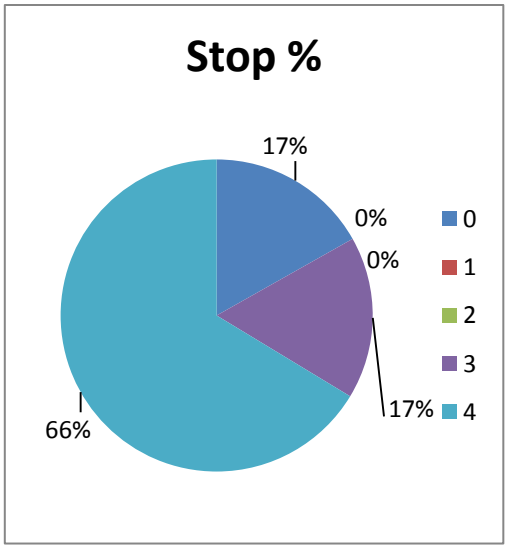


Figure 5.8.6

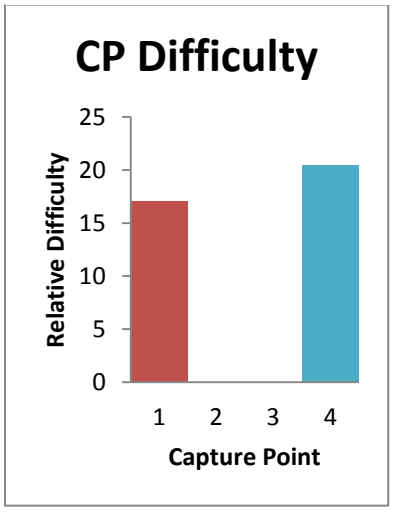
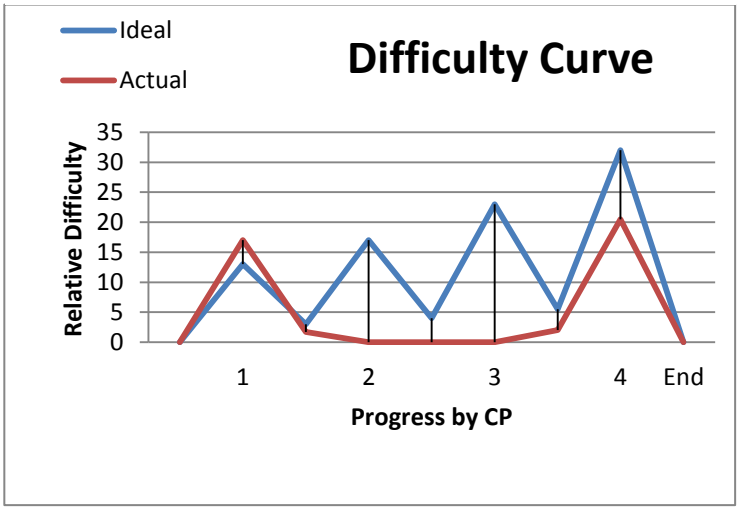


Figure 5.8.7



5.9: Iteration 9

Version: pl_frontier_a10(Alpha 10)

Contextually detailed prototype

Changelog:

- Path into rock structure before segment 1 choke again rerouted, this time dropping down next to the ramp added into the building it used to connect to.
- Dropdown in segment 2 hill building moved to opposite side.
- Train speed decreased once more.

Figure 5.9.1

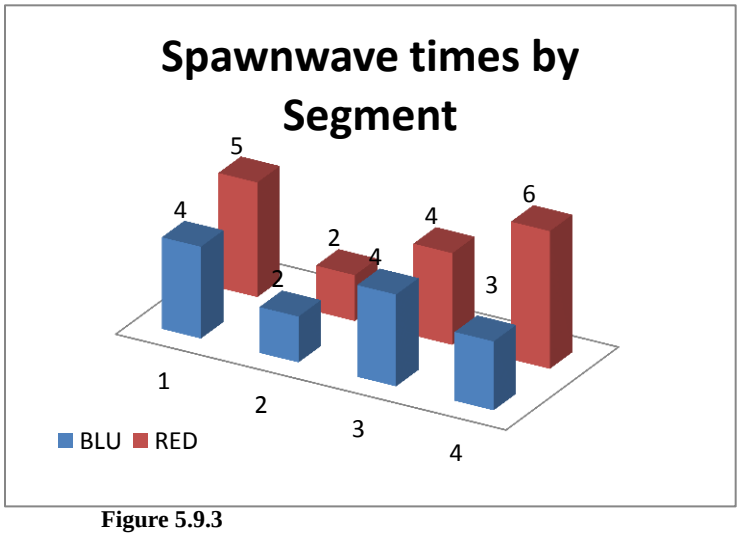


Figure 5.9.2

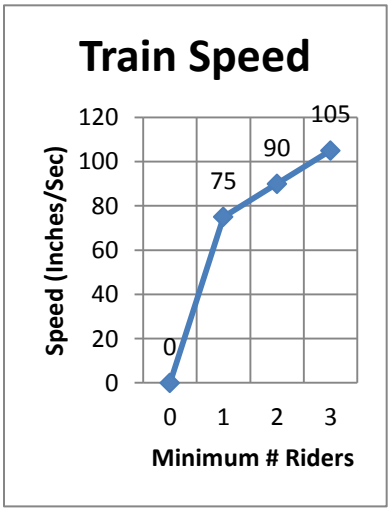
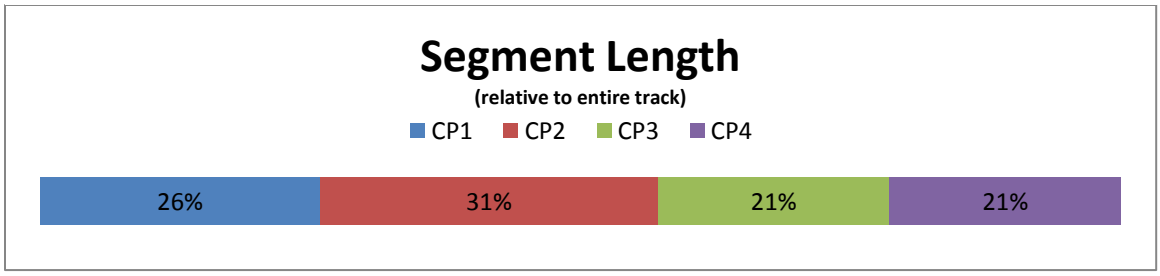


Figure 5.9.3



Iteration 9 Playtest Results Summary:

325 Unique Playtesters

61% Offensive Wins

81% of Map Experienced on Average

69% Deviation from Ideal Curve

Figure 5.9.4: Player Death Heatmap

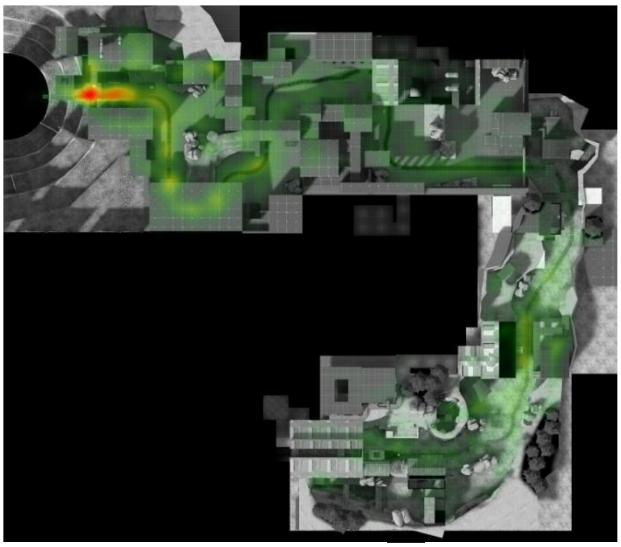


Figure 5.9.5

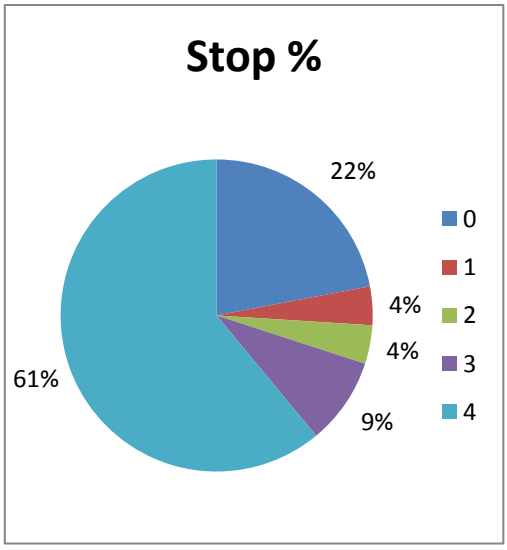


Figure 5.9.6

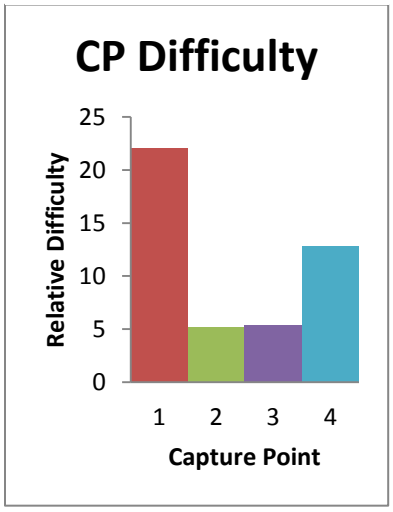
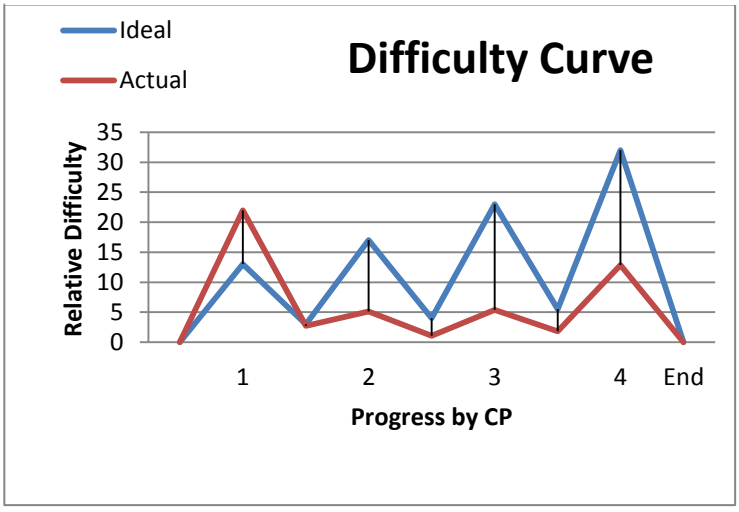


Figure 5.9.7



5.10: Iteration 10

Version: pl_frontier_a11 (Alpha 11)

Contextually detailed prototype

Changelog:

- No significant layout changes were made.
- Gameplay-impacting lipping errors from a10 corrected.

Figure 5.10.1

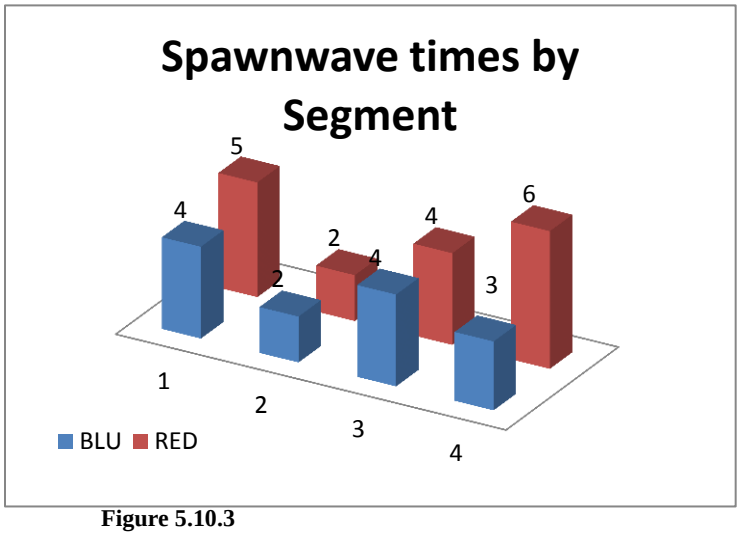


Figure 5.10.2

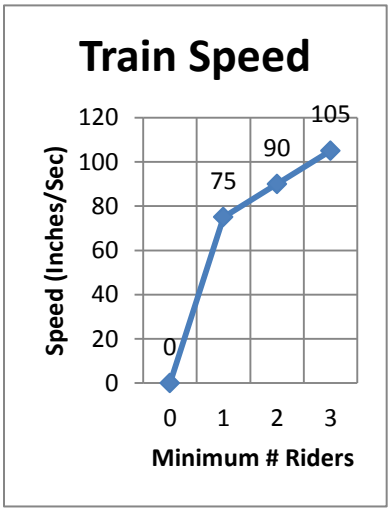
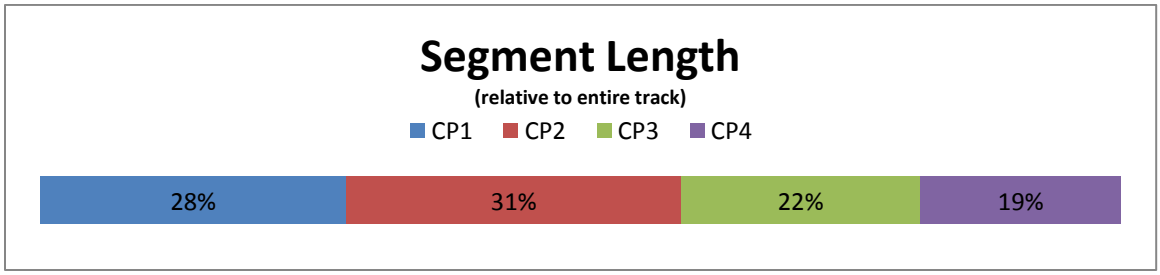


Figure 5.10.3



Iteration 10 Playtest Results Summary:

329 Unique Playtesters

48% Offensive Wins

70% of Map Experienced on Average

103% Deviation from Ideal Curve

Figure 5.10.4: Player Death Heatmap

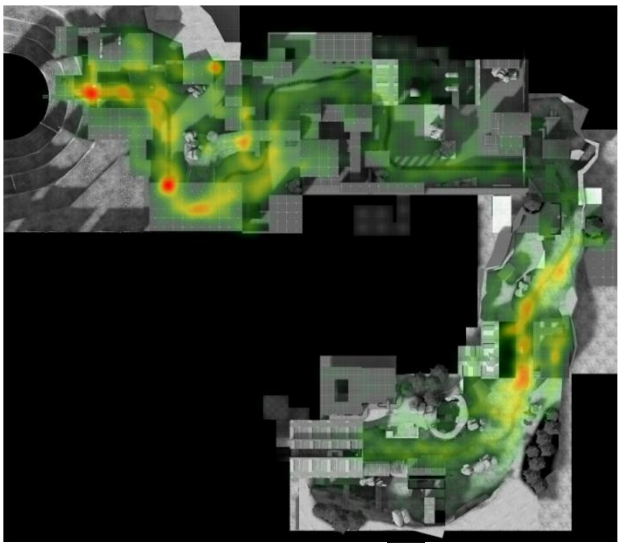


Figure 5.10.5

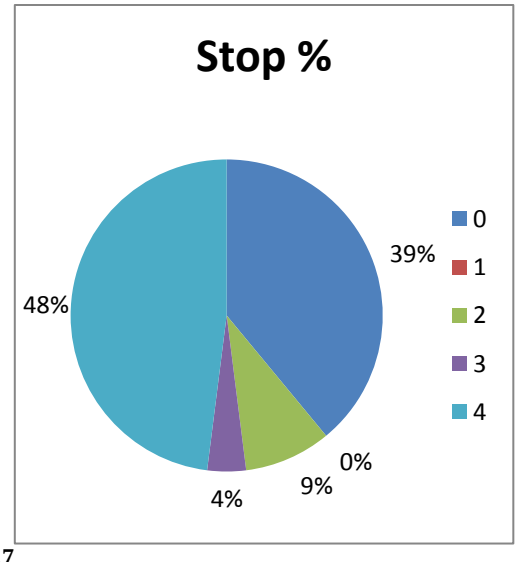


Figure 5.10.6

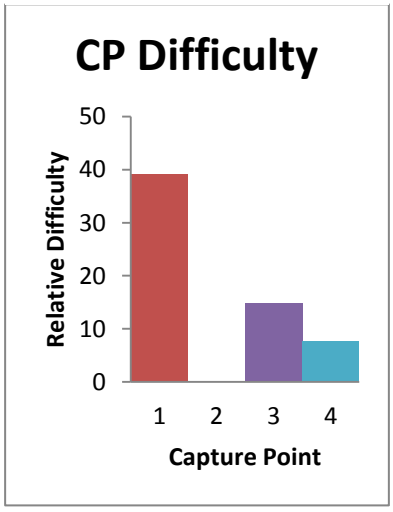
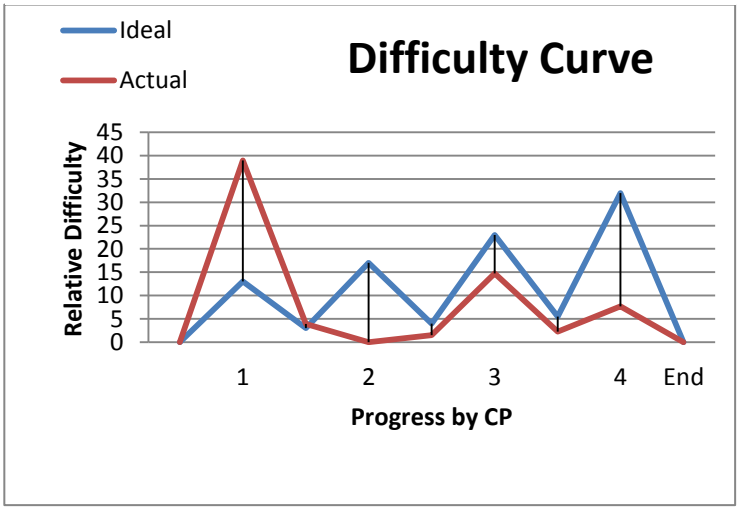


Figure 5.10.7



5.11: Iteration 11

Version: pl_frontier_a12 (Alpha 12)

Contextually detailed prototype

Changelog:

- No significant layout changes were made.
- Train speed increased by a factor of 1.3 after CP3 is captured, effectively shortening segment 4 by a factor of ~23%.
- RED spawnwave time increased for segment 2 and decreased slightly for segment 4.

Figure 5.11.1

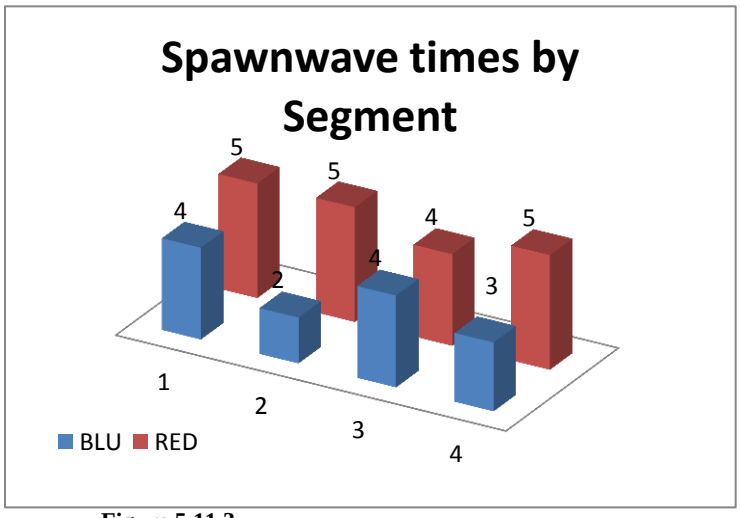


Figure 5.11.2

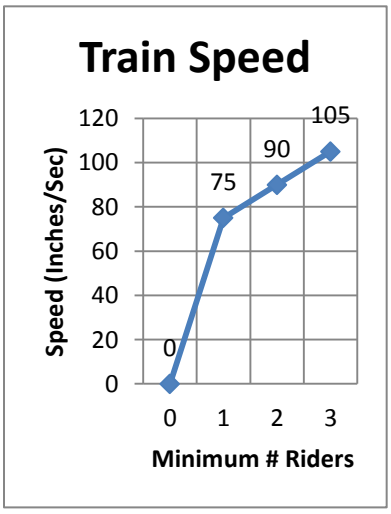
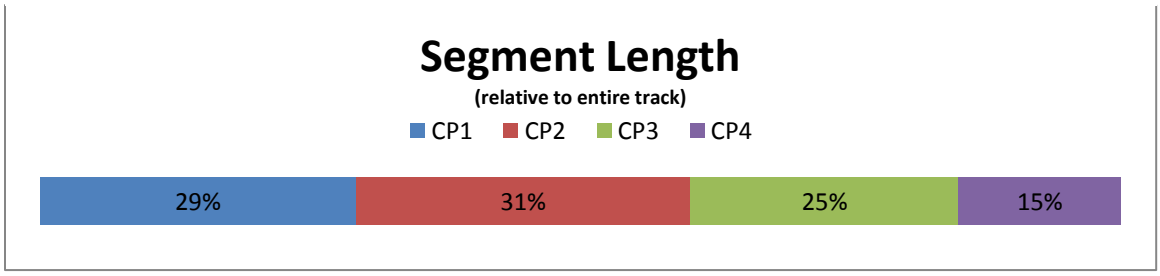


Figure 5.11.3



Iteration 11 Playtest Results Summary:

163 Unique Playtesters

50% Offensive Wins

77% of Map Experienced on Average

60% Deviation from Ideal Curve

Figure 5.11.4: Player Death Heatmap

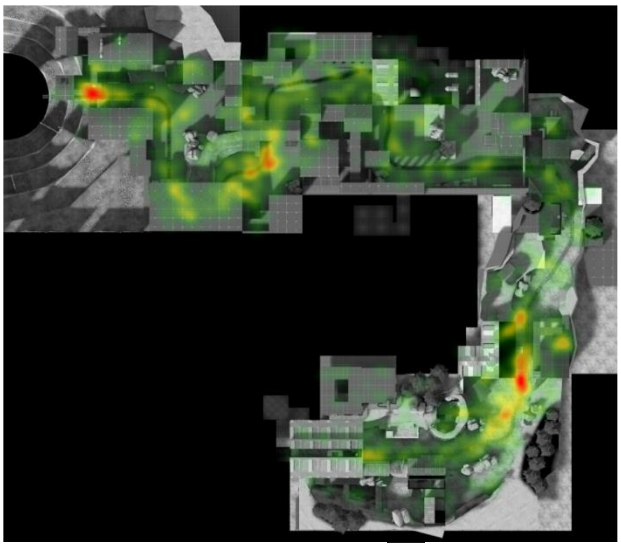


Figure 5.11.5

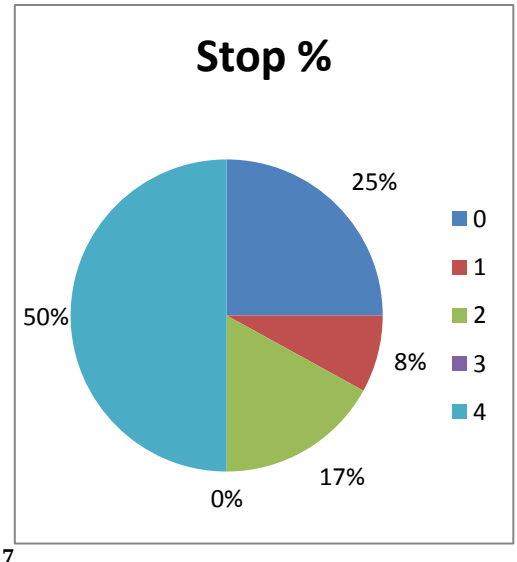


Figure 5.11.6

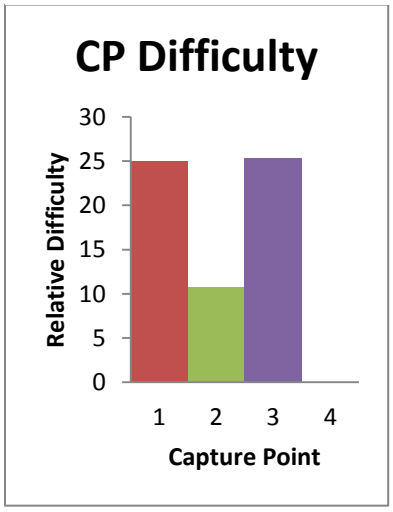
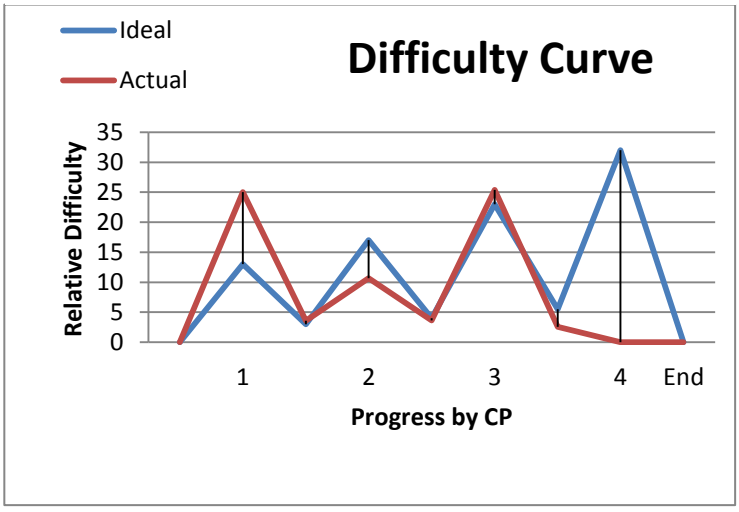


Figure 5.11.7



5.12: Iteration 12

Version: pl_frontier_a13 (Alpha 3)

Contextually detailed prototype

Changelog:

- No significant layout changes were made.
- Changes to RED spawn wave time during segment 2 reverted and decreased further.
- BLU spawn wave time reduced to 0 during segment 3.

Figure 5.12.1

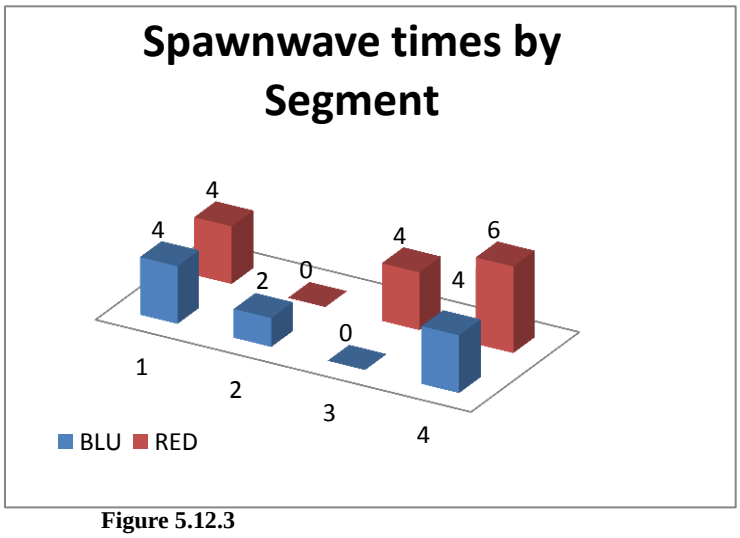


Figure 5.12.2

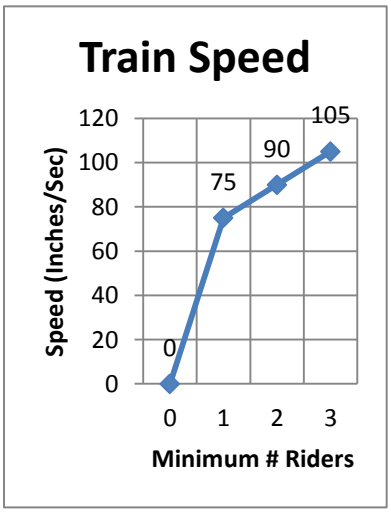
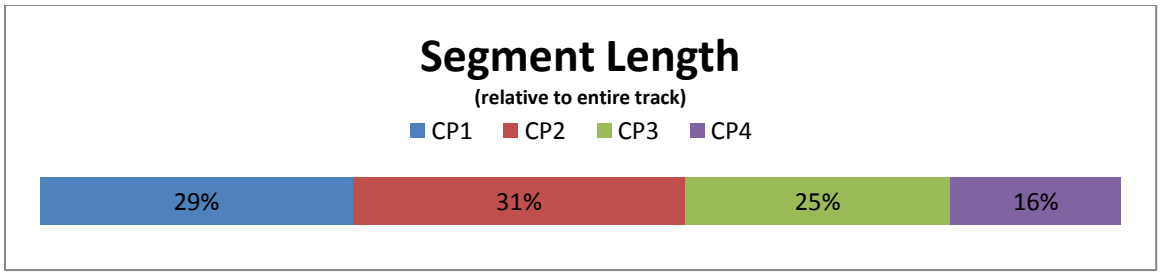


Figure 5.12.3



Iteration 12 Playtest Results Summary:

1711 Unique Playtesters

39% Offensive Wins

69% of Map Experienced on Average

75% Deviation from Ideal Curve

Figure 5.12.4: Player Death Heatmap

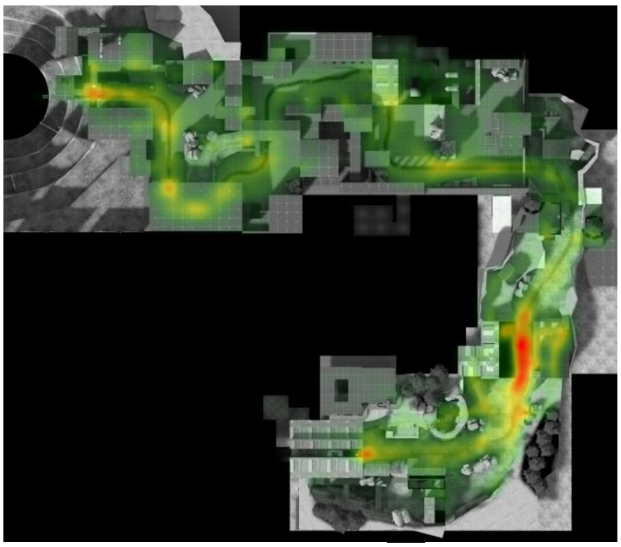


Figure 5.12.5

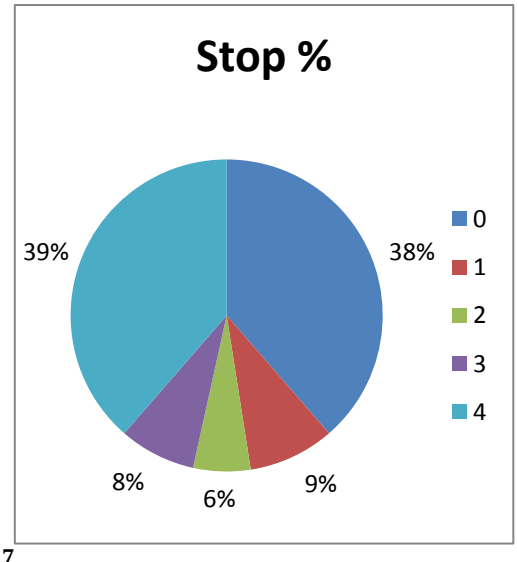


Figure 5.12.6

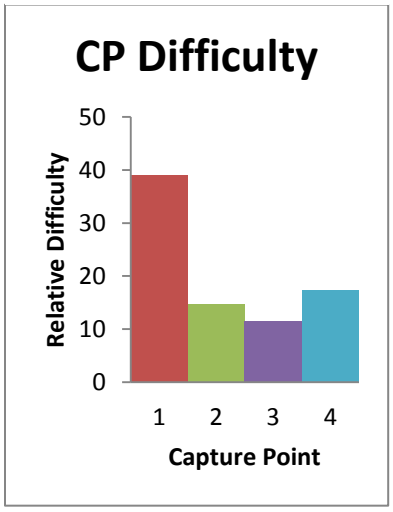
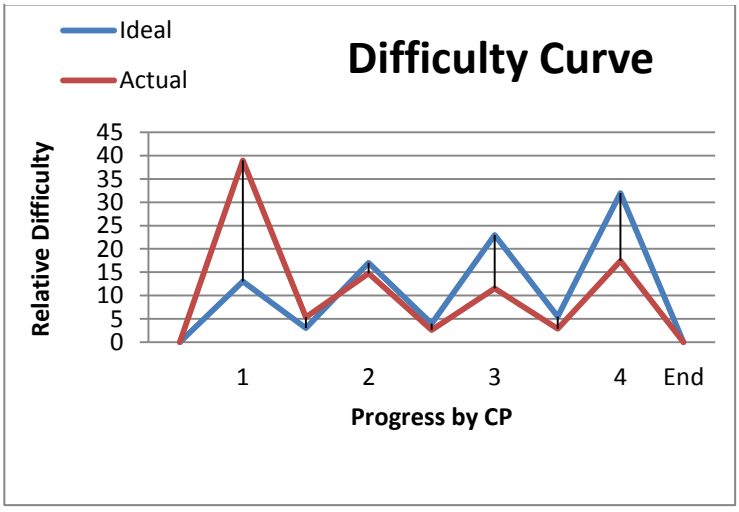


Figure 5.12.7



Chapter 6

Extended Playtest Decisions & Results

It is important to distinguish the rapid release schedule of the alpha-build stage from the slower and more cautious beta testing phase. Having reached a point where overall layout is satisfactory, an art pass may be applied to fully realize the level within the context of the game story and environment. Deciding upon that stop point is not a precise science, but with further playtesting available ahead it can be up to the level designer's discretion to choose without undoable consequence.

After an art pass is performed, each new change to layout can be much more costly in terms of development time. Each new iteration henceforth is tested for a longer period of time, to better inform changes performed on its successor.

Again, please reference **Appendix B** for the equations used on the data in **Appendix C** in generating the following graphs and figures.

6.1: Iteration 13

Version: pl_frontier_bc (Beta Candidate)

Detailed, unlit level; art pass changes tested before beta release

Changelog:

- First art pass applied. Innumerable small geometry and clipping changes made as a result.
- Cart replaced with first 'chewchew' model.
- Leftside exit of first BLU spawn restructured to provide height advantage over battlefield; pathway towards segment 1 choke building removed.
- Bridges spanning puddle to right of first BLU spawn removed.
- Accessible window in segment 1 choke building moved to the left to limit sentry dominance.
- Ramp added from first RED spawn onto rooves overlooking segment 1 choke.
- Path from segment 1 cliff to segment 2 hill completely removed.
- Bridge structure preceding segment 2 hill stripped of cover.
- Interior of CP3 warehouse adjusted to provide greater access to catwalks.
- Train speed again decreased by about 20%.
- Defensive spawn times increased, offensive decreased.

Figure 6.1.1

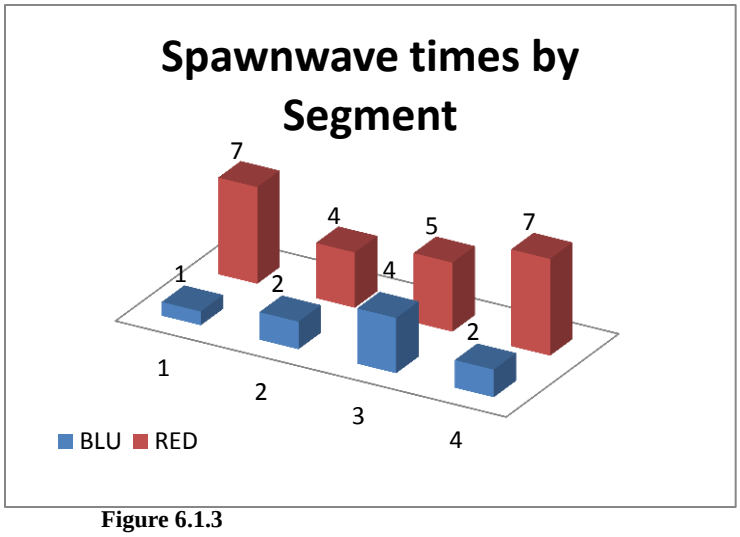


Figure 6.1.2

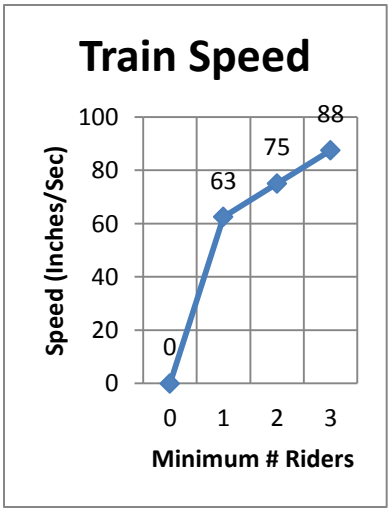
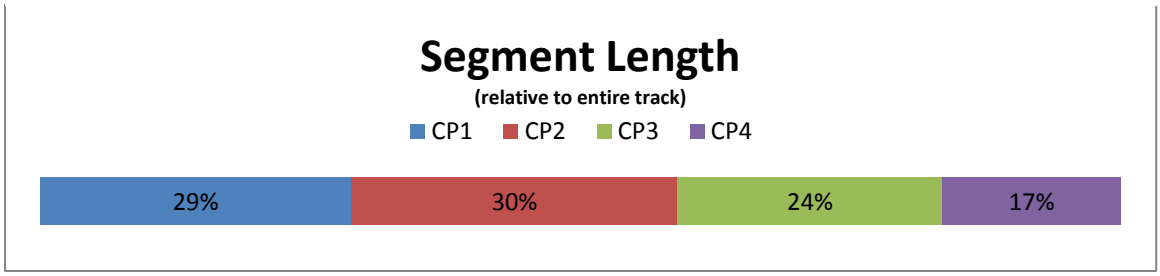


Figure 6.1.3



Iteration 13 Playtest Results Summary:

658 Unique Playtesters

51% Offensive Wins

75% of Map Experienced on Average

65% Deviation from Ideal Curve

Figure 6.1.4: Player Death Heatmap

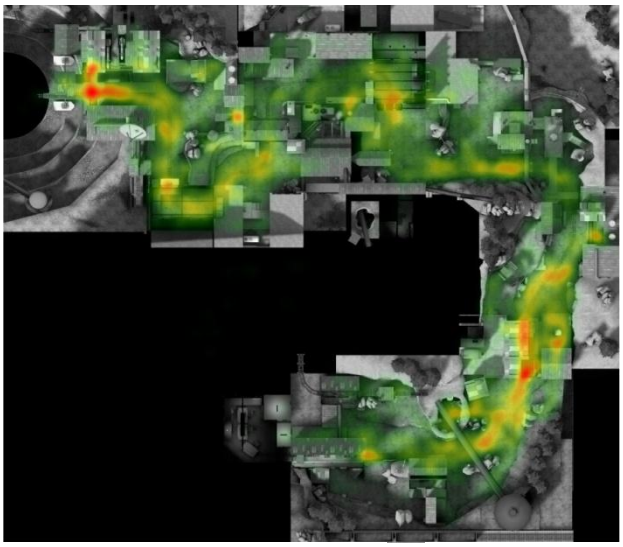


Figure 6.1.5

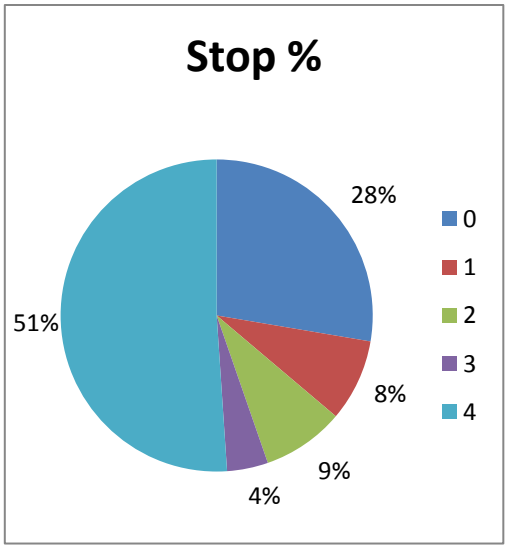


Figure 6.1.6

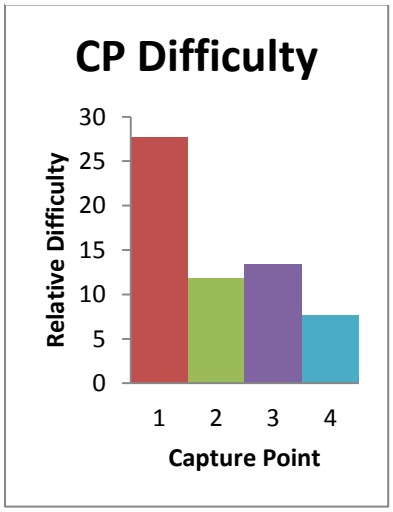
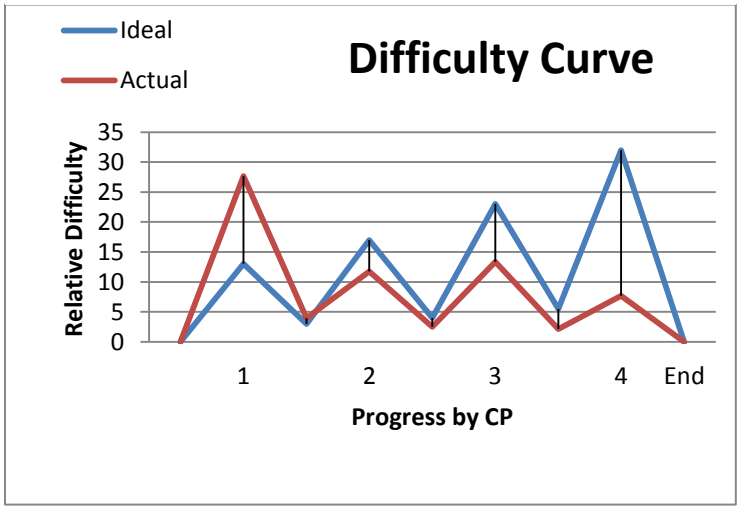


Figure 6.1.7



6.2: Iteration 14

Version: pl_frontier_b1 (Beta 1)

Detailed lit level

Changelog:

- Opimization performed, unaffecting layout geometry significantly.
- Lighting pass run.
- Slight reduction to RED spawn times on latter half of map.

Figure 6.2.1

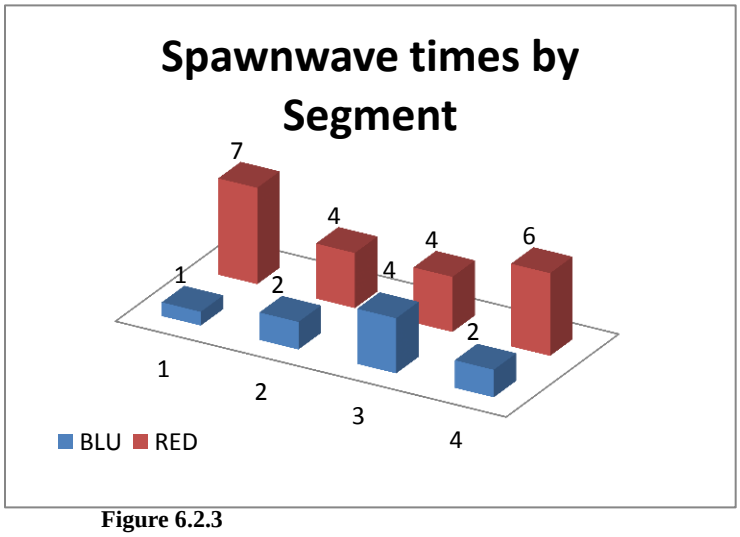


Figure 6.2.2

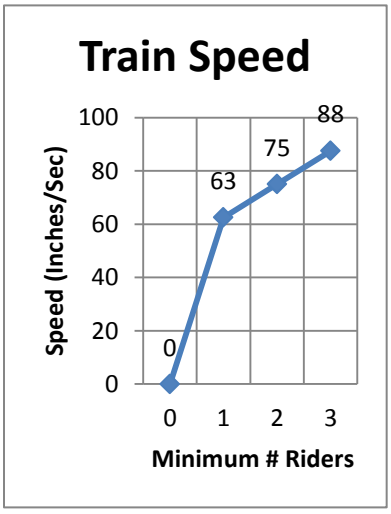
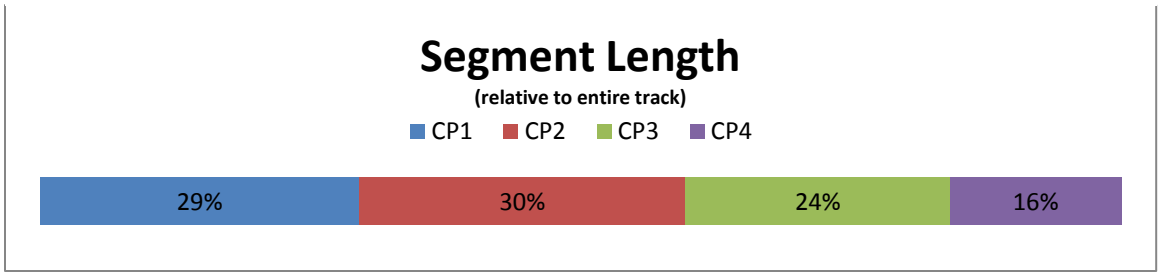


Figure 6.2.3



Iteration 14 Playtest Results Summary:

1956 Unique Playtesters

41% Offensive Wins

72% of Map Experienced on Average

58% Deviation from Ideal Curve

Figure 6.2.4: Player Death Heatmap

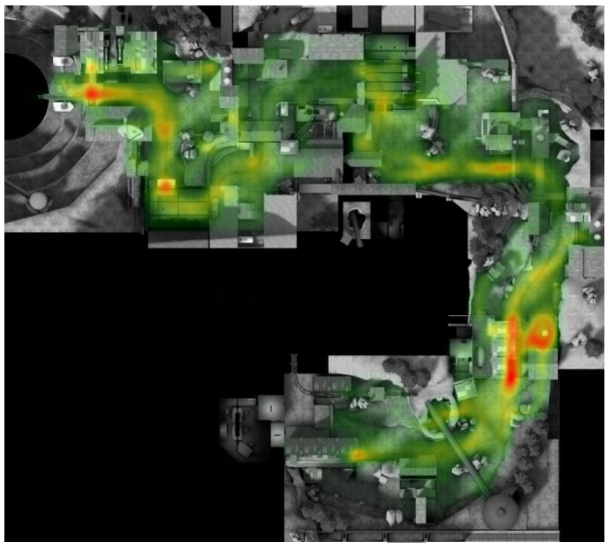


Figure 6.2.5

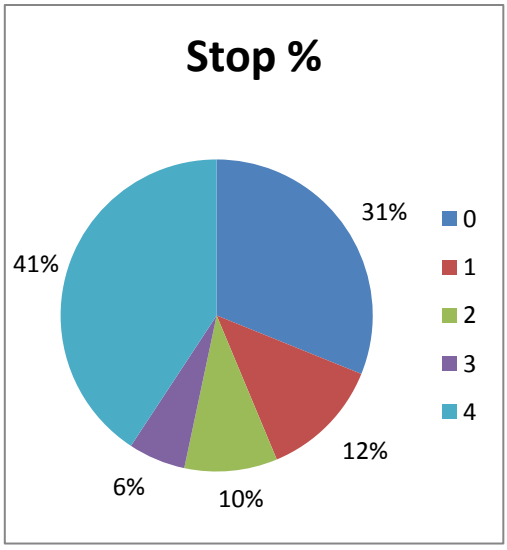


Figure 6.2.6

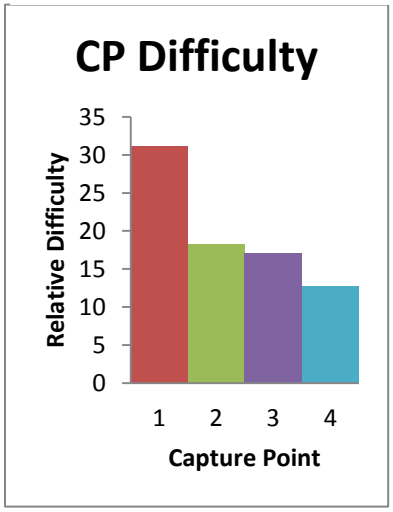
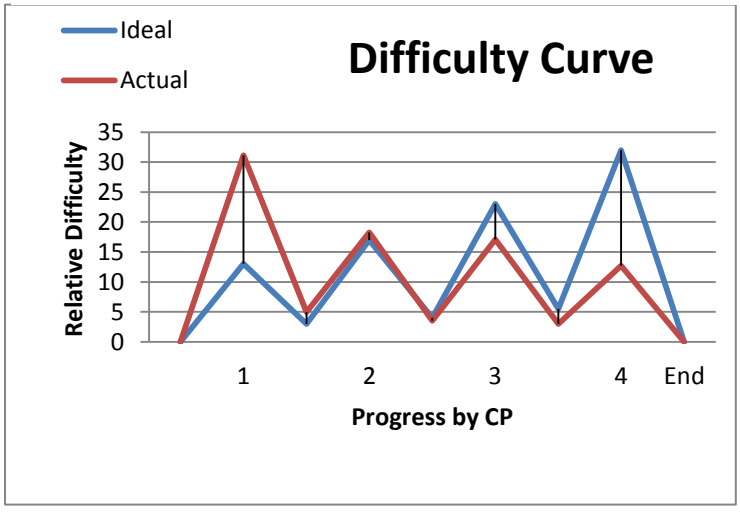


Figure 6.2.7



6.3: Iteration 15

Version: pl_frontier_b2 (Beta 2)

Detailed lit level

Changelog:

- Cart replaced with second 'chewchew' model; ramming ability changed to biting ability (no gameplay impact.)
- Various small geometry changes to help with optimization.
- Interior of segment 1 choke building completely removed for optimization and gameplay reasons.
- Path added onto roof of aforementioned building. Attic area joined onto roof to provide cover.
- Two exits of segment 1 cliff/choke after CP1 building removed for optimization reasons.
- Dropdown from segment 2 hill building moved back to original position.
- BLU spawn time for segment 1 reduced to 0 to match Valve standards.

Figure 6.3.1

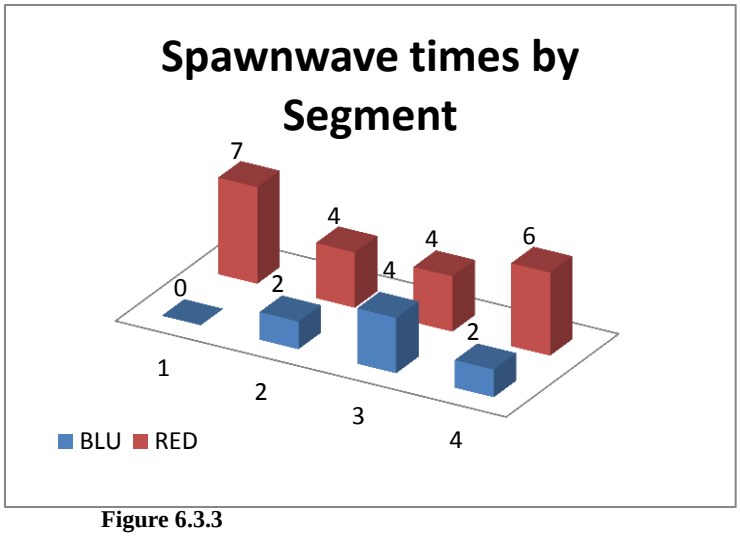


Figure 6.3.2

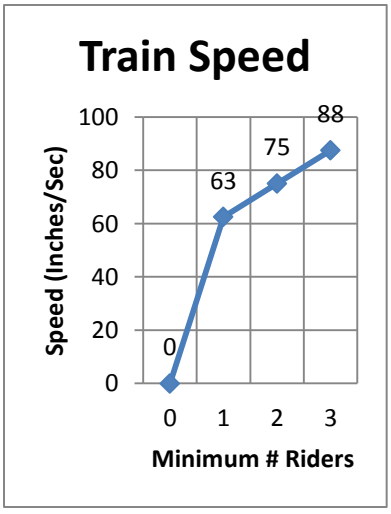
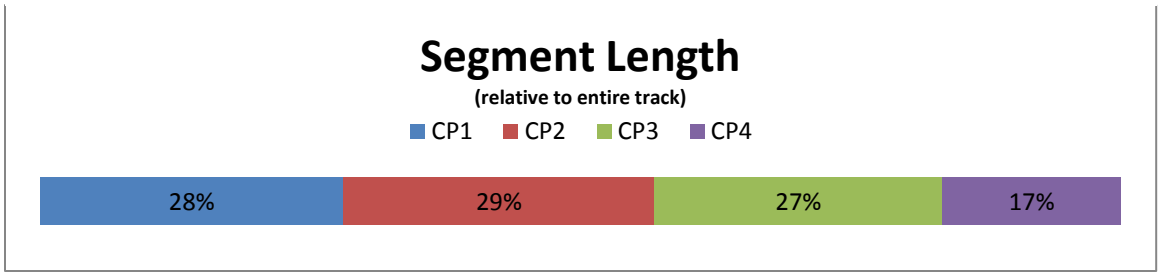


Figure 6.3.3



Iteration 15 Playtest Results Summary:

5151 Unique Playtesters

45% Offensive Wins

79% of Map Experienced on Average

31% Deviation from Ideal Curve

Figure 6.3.4: Player Death Heatmap

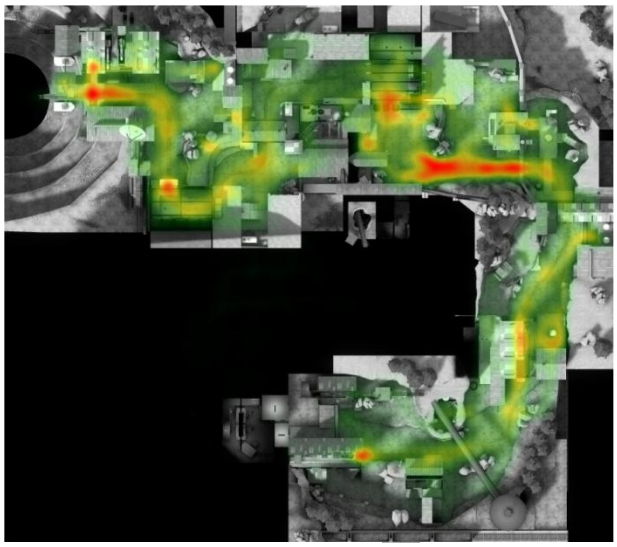


Figure 6.3.5

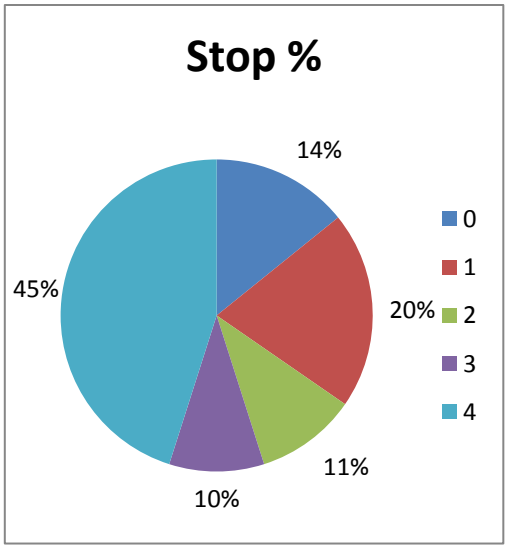


Figure 6.3.6

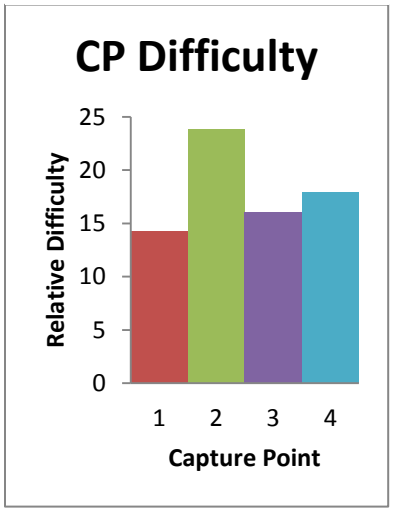
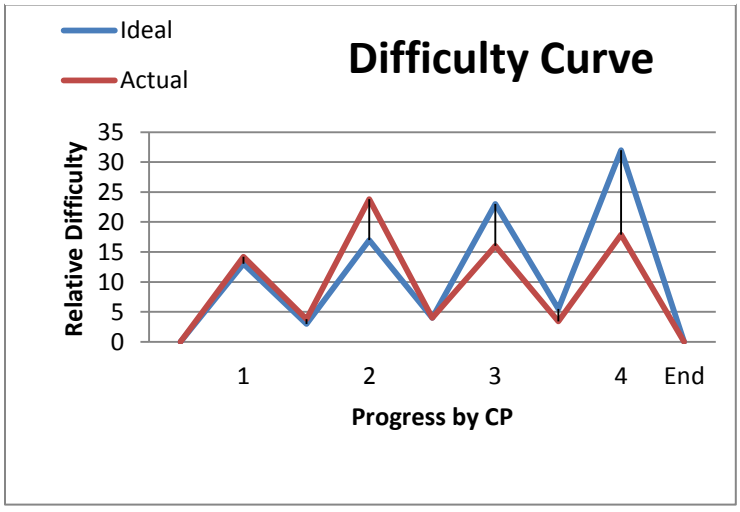


Figure 6.3.7



6.4: Iteration 16

Version: pl_frontier ('Finished' map released to public for broad testing; Release Candidate)

Detailed lit level

Changelog:

- Dropdown in segment 2 hill building completely removed.
- Small geometry, optimization and clipping errors corrected.
- Train speed boosted up 13%.
- Final point position adjusted slightly.

Figure 6.4.1

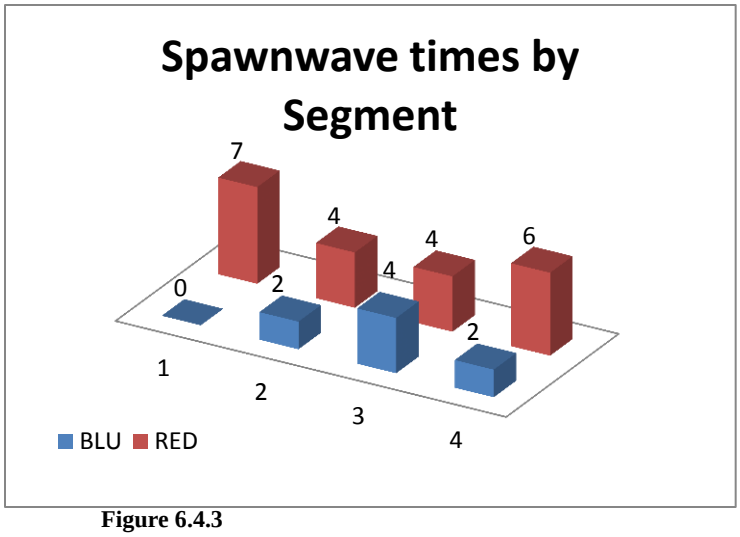


Figure 6.4.2

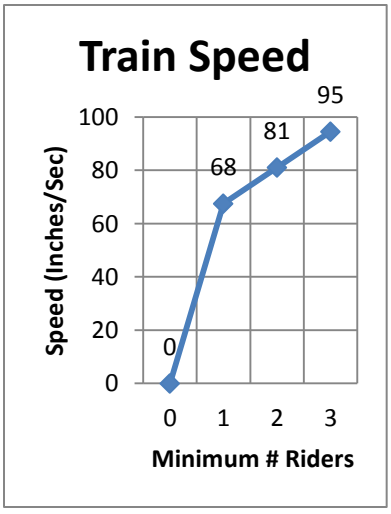
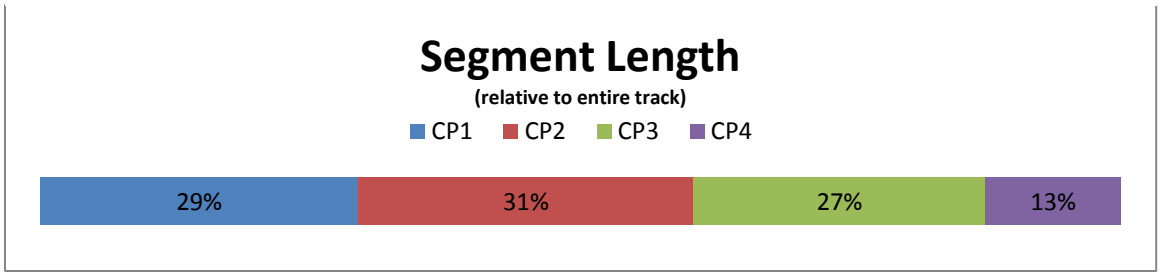


Figure 6.4.3



Iteration 16 Playtest Results Summary:

8632 Unique Playtesters

45% Offensive Wins

82% of Map Experienced on Average

23% Deviation from Ideal Curve

Figure 6.4.4: Player Death Heatmap

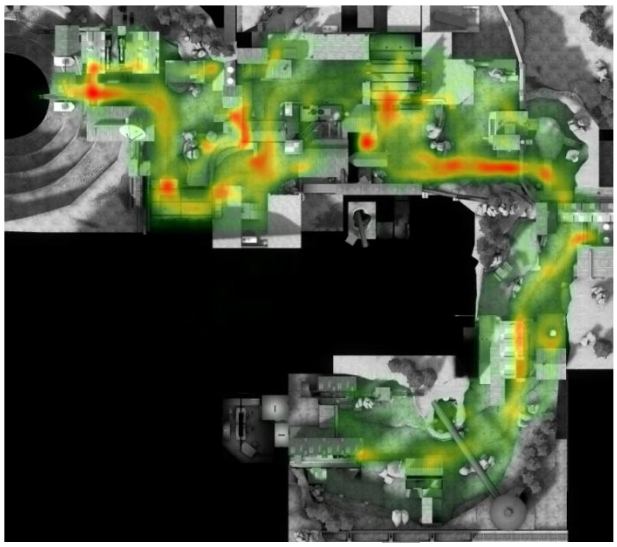


Figure 6.4.5

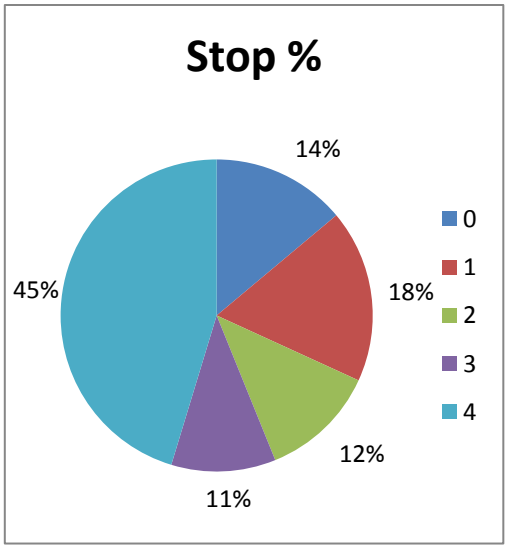


Figure 6.4.6

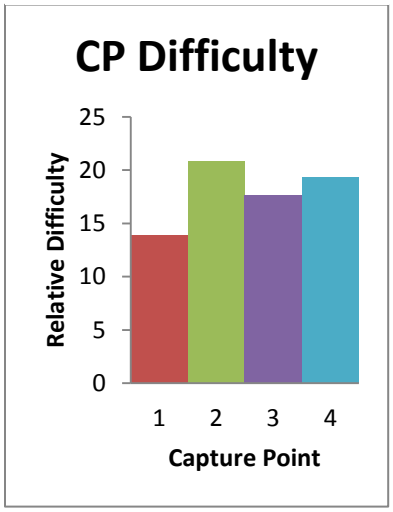
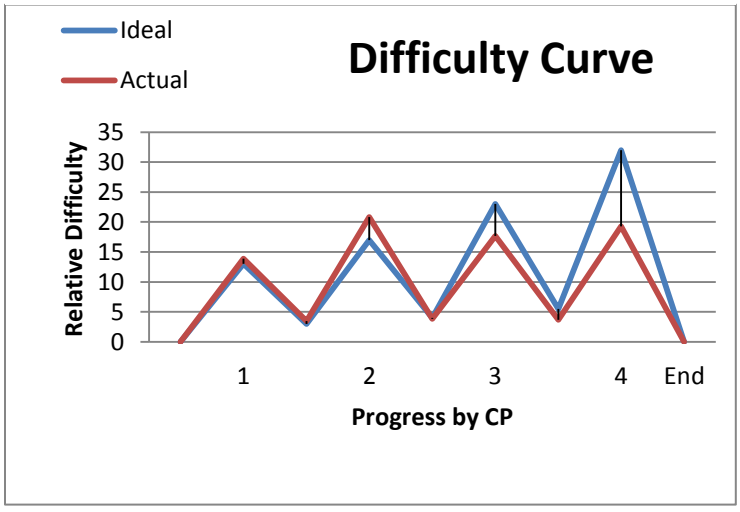


Figure 6.4.7



6.5: Iteration 17

Version: pl_frontier_final (Final version shipped by Valve to every player)

Fully polished level

Changelog:

- Vastly improved optimization; Lighting improved universally.
- Changed all disappearing brushes used as gates into proper sliding doors.
- First BLU spawn interior rearranged to provide better cover for offense within.
- Flow around bottom of segment 2 hill adjusted to offense's benefit. Lower area to right of second hill entrance removed completely.
- Flow within CP2 warehouse altered to better accommodate usual player activity inside.
- Path from side of CP2 warehouse around and above to segment 3 curved into spiral staircase to protect players and better optimize.
- Ground immediately out of former warehouse CP2 raised and hut added to provide cover for offensive players.
- Path from segment 3 rock tiers into warehouse made shorter to offense's benefit.
- Impenetrable window added as cover / spam blocker in front of shortcut from segment 2 to segment 4.
- Health and ammo pickup locations rearranged around last point to defense's advantage.

Figure 6.5.1

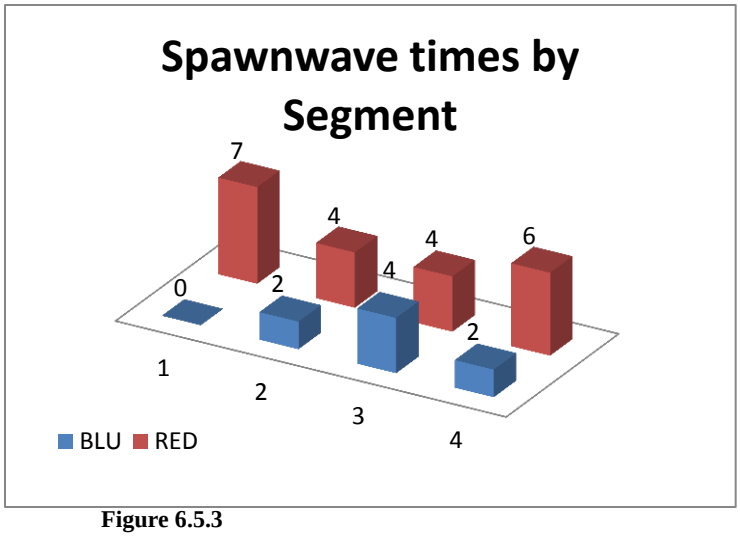


Figure 6.5.2

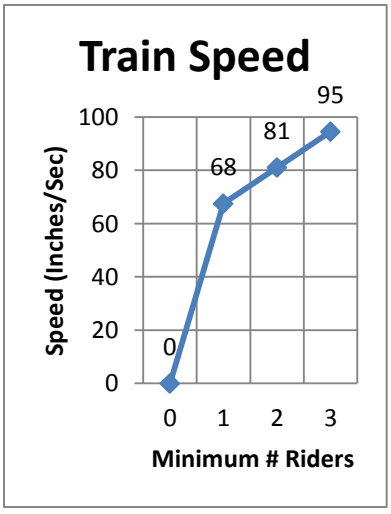
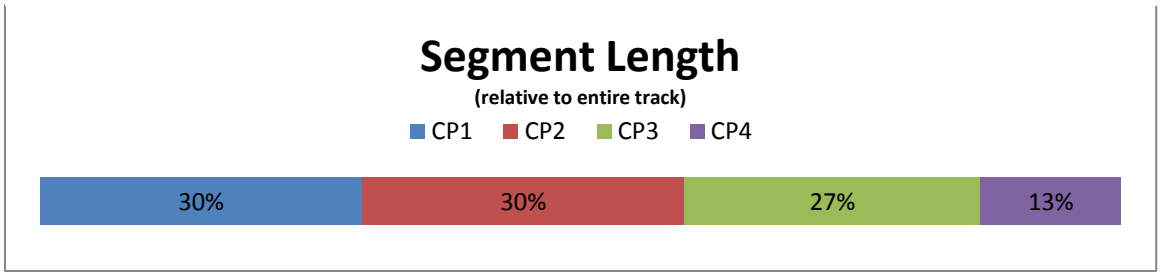


Figure 6.5.3



Frontier Gameplay Metrics Summary:

562 Unique Playtesters

44% Offensive Wins

86% of Map Experienced on Average

10% Deviation from Ideal Curve

Figure 6.5.4: Player Death Heatmap

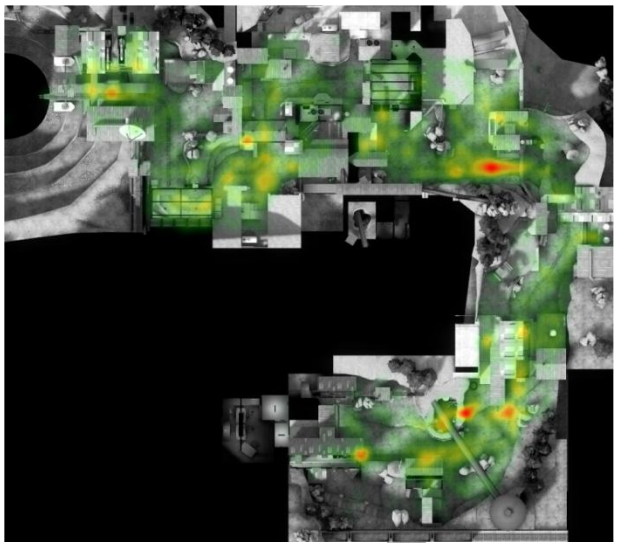


Figure 6.5.5

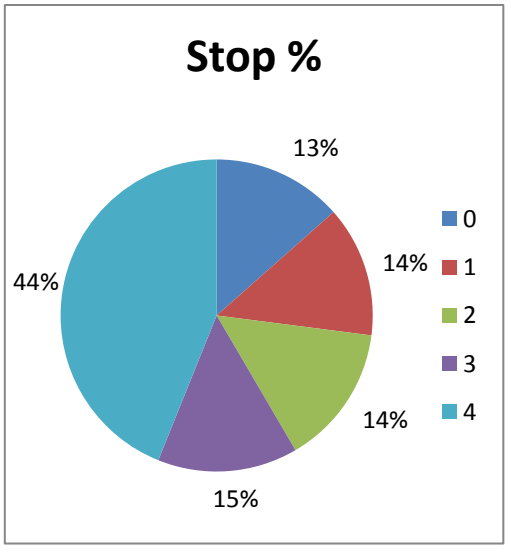


Figure 6.5.6

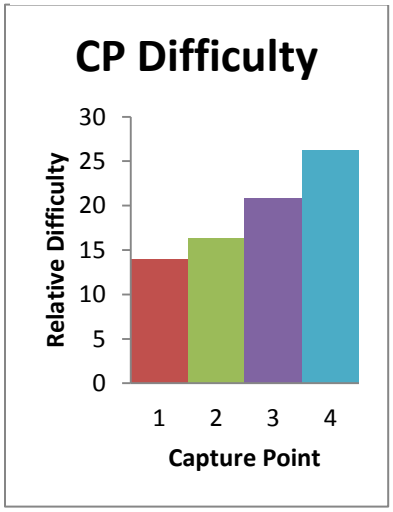
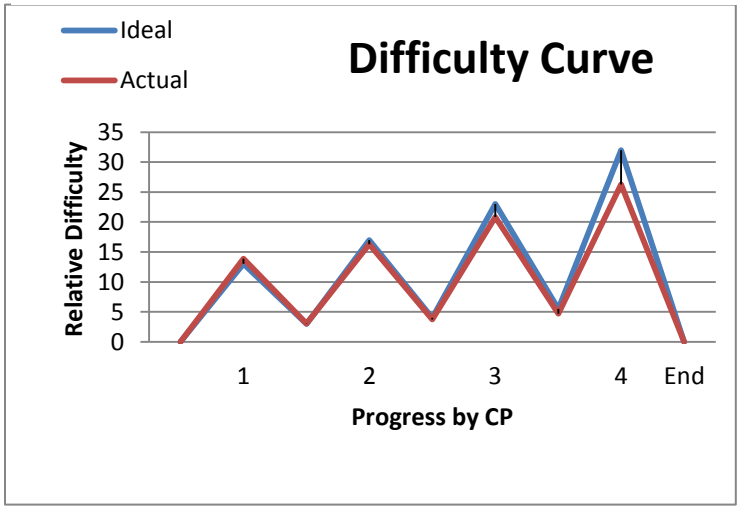


Figure 6.5.7



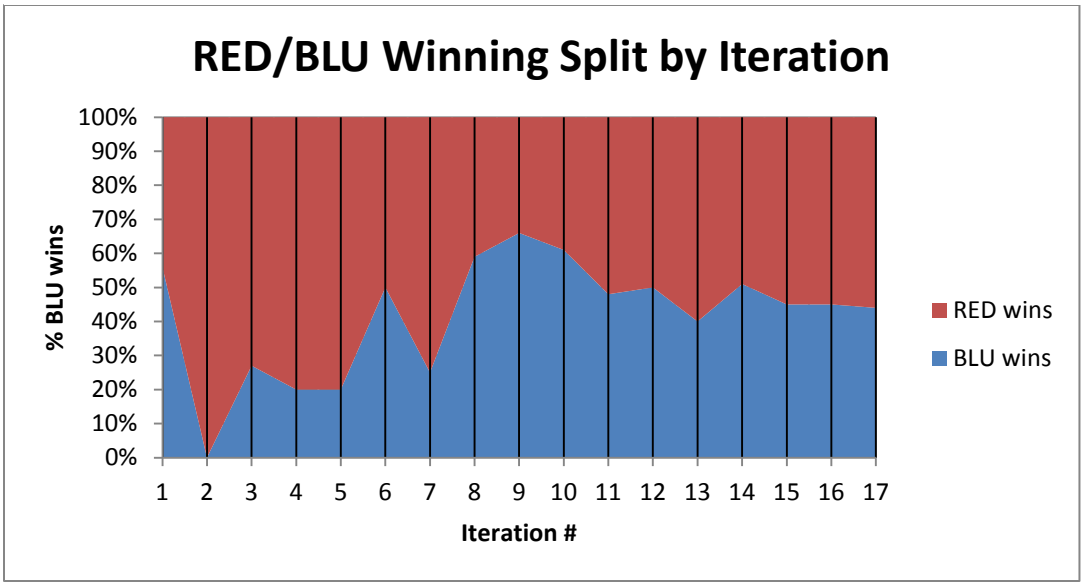
Chapter 7

Concluding Notes & Takeaway

7.1: Summary of Findings

It is helpful to look at the evolution of the map over its entire development schedule as a whole. To this end, the statistics summarizing each iteration have been compiled. In a practical sense, this information would be analyzed in between each iteration for the reaching of predetermined development stop conditions. Individual stop conditions need not be overly specific: as all factors in a map are confounding to some degree, a judgment of conditions as a whole is necessary. Still, Badwater's own stats provide a good baseline for decision making.

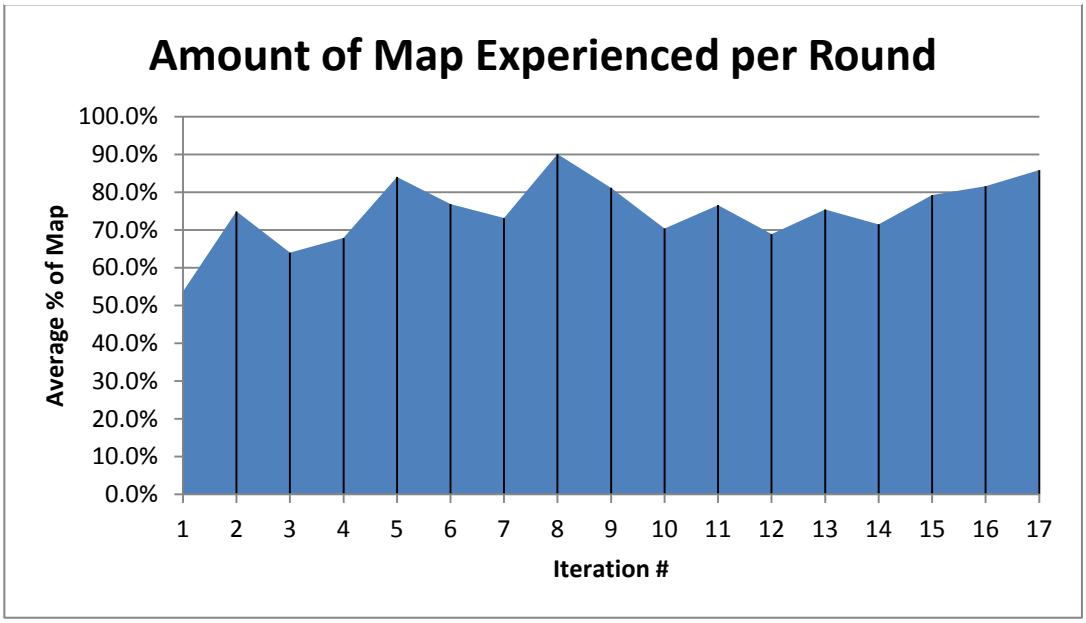
Figure 7.1.1



The first and most immediately interpretable of our goals / stop conditions was the win split between the offense and the defending team. This figure gauges overall fairness of competition between the two teams of players; it is of note that the winning rate of the teams are

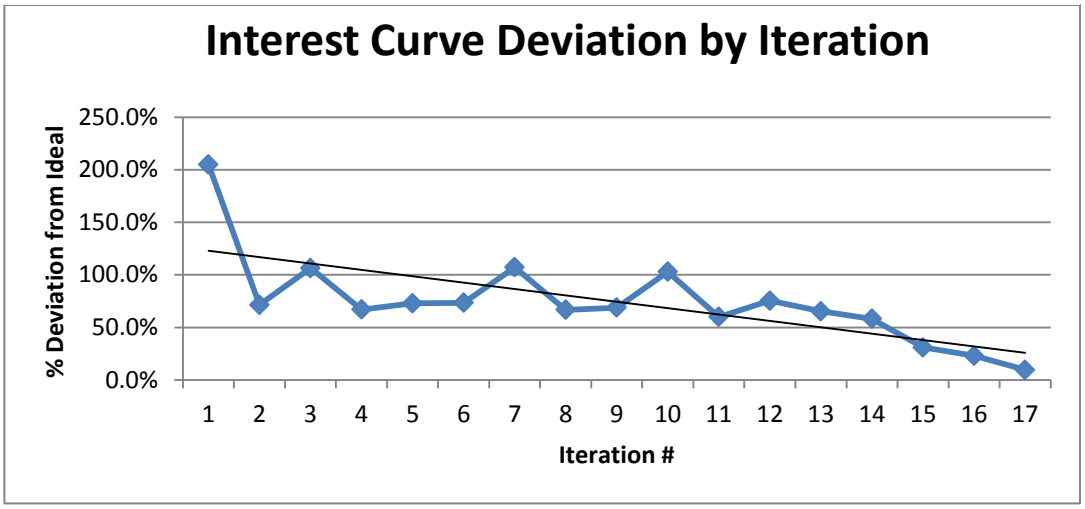
merely averaged over all plays of the map, and also that each teams' rates are complementary. As the above graph displays, the winning split jumped wildly during the first, rapid segment of development (through iteration 10.) However, later revisions helped to stabilize the ratio and bring it closer to the ideal 50% point—a satisfactory fulfillment of this stop condition. Furthermore, as Badwater possesses an overall 39% win ratio, Frontier's final 44% might be considered superior.

Figure 7.1.2



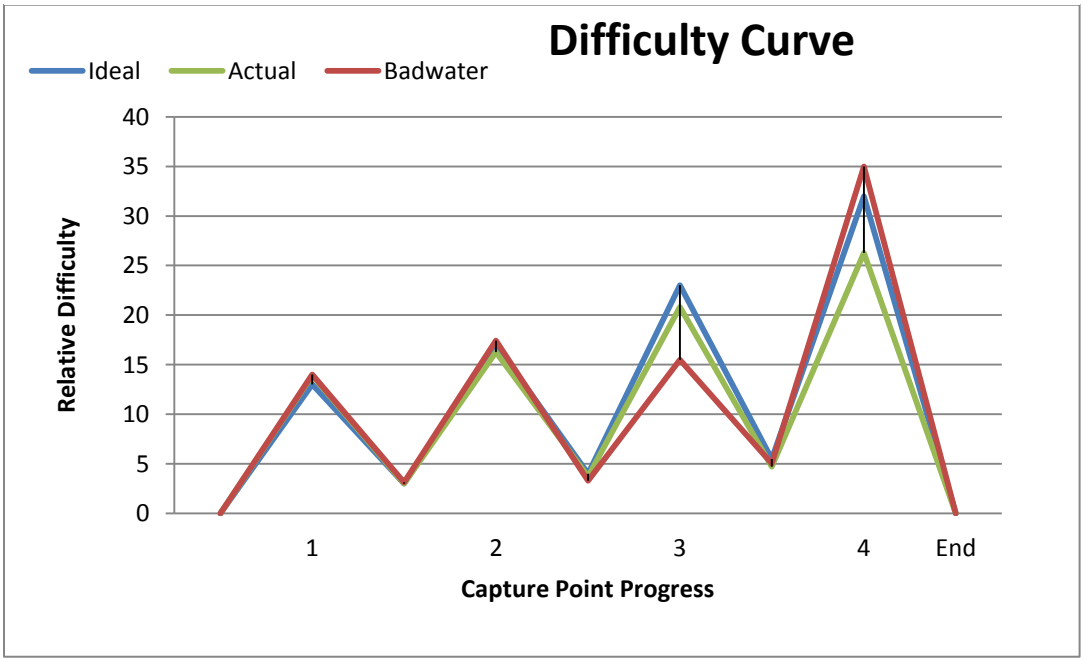
A second goal was the maximization of developer effort : player utility by ensuring as large a percentage of the map is played, or experienced, on average each round. The above graph shows a trend similar to that of the **Winning Rate's**: instability during the rapid prototyping session, with a smoothening occurring around iteration 12. At that point it gradually builds up over each version until reaching a suitable 86%. Of course, this number could be made higher—but lowering the difficulty of early CP segments to do so would wreck the map's Interest Curve. With Badwater's respective figure at 83% in this area too might Frontier be equal or greater.

Figure 7.1.3



The map's Interest Curve represented our closest quantifiable equivalent to its enjoyability factor. Reducing the deviation of the curve from our formerly stated ideal was one goal towards improving the map's gameplay. While each segment's difficulty varied greatly from test to test as alterations were tested, the curve's deviation as a whole gradually sloped downwards over the course of development. From a high of 205% the interest curve was steered

Figure 7.1.4



towards a quite acceptable 10% deviation in the final product. As with the previously graphed goals, rapid prototyping saw alternating peaks of deviation, but after the art pass the rate of decreasing deviance stabilized to a more predictable state. This figure, compared to Badwater's deviation of 13%, demonstrates how this way of guiding iterative decisions can directly impact the interest curve.

7.2 Conclusion

Having satisfied all 3 stop goals to an acceptable degree—equaling or surpassing the control, Valve's own Badwater, on each criterion—the map has been declared final for the moment.

Of course throughout Frontier's development the relationship between an ideal interest curve and player gratification had been theoretical and assumed. With the final map released as an official map in TF2 there now exists data suggesting its popularity relative to the other maps in the game, thus providing evidence for that relationship's strength. As it sits the month of April 2011, Frontier ranks 9th most popular among all official maps within TF2 out of the 46 maps included within the game at that time. Of further note is that it is the most popular out of all user-made (i.e. community) maps bought by Valve, with over 22,000 rounds played on it globally.

Whether this stands as definitive proof of this particular iterative process' effectiveness on a map's fun factor may be debatable, but as none of the other community maps employed such empirical methodology it was quite likely a large factor in steering the development process.

7.3 Takeaway

The takeaway for a level designer having read this paper ought to be the following (generalized) process proven valid by the case study's success:

1. Identify metrics of gameplay within a map that hold a relationship to solid level design principles.
2. Set goals within those metrics as development stop points to work towards.
3. Manipulate level layout and relevant independent variables to progressively maneuver the selected metrics towards their respective stop points.
4. Track progress through the development cycle to observe gameplay trends in regards to those stop points.
5. Finally, evaluate the resultant user experience through playtesting observation to prove the success of this guided process of iteration's.

More specifically, using this process in the context of shaping a level's (or a series of levels') interest curves should be a new concept to readers—one pursuable in their next project's creation, to ensure an "epic-feeling" gameplay experience for the end user.

References

Allen, C. (Speaker). (2007). *Map Prototype Process: Creating Fun Gameplay Spaces While Minimizing Risks* [MP3]. GDC Vault.

Ambinder, M. (Speaker). (2009). *Valve's Approach to Playtesting: The Application of Empiricism* [MP3]. GDC Vault.

Andruko, C., & Everman, D. (Speakers). (2009). *The Iterative Level Design Process of Bioware's MASS EFFECT 2* [MP3]. GDC Vault.

Antonov, V., Watkins, R., & Olcayto, R. (Speakers). (2009). *Edge Panel: Architecture and Videogames* [MP3]. Retrieved from http://www.next-gen.biz/features/edge-panel-architecture-and-videogames

Co, P. (2010, March 19). 12 Terms from the Left 4 Dictionary [Web log post]. Retrieved from Left 4 Dead Blog: http://www.l4d.com/blog/post.php?id=3623

Corporation, V. (Speaker). (2007). *Half-Life 2: Lost Coast, Episode 2, and Episode 2 Developer Commentary* [Video Game].

Corporation, V., & Walker, R. (Speakers). (2007). *Team Fortress 2 Developer Commentary* [Video Game].

Jacques-Belletete, J. (Speaker). (2010). *Creating a Unique Visual Direction: The Successes and Failures of Creating a Near-Future Cyberpunk Setting with a Renaissance Twist in Deus Ex* [MP3]. GDC Vault.

Johnson, T. (2010, February 22). Stalling Players with Gates [Web log post]. Retrieved from http://www.nodraw.net/2010/02/stalling-players-with-gates/

Jungels, J. (2008, August 1). TF2 Environments [Web log post]. Retrieved from TF2 Offical Blog: http://www.tf2.com/post.php?id=1729&p=9

Kremers, R. (2009). *Level Design: Concept, Theory & Practice*. A.K. Peters, Ltd.

Lundeen, R. (2009, January 2). L4D Art Direction, Part 2: Stylized Darkness [Web log post]. Retrieved from Left 4 Dead Blog: http://www.l4d.com/blog/post.php?id=2129

Morin, J. (Speaker). (2009). *Player's Expression: The Level Design Structure Behind FAR CRY 2 and Beyond* [MP3]. GDC Vault.

Mulholland, P., & Keranen, I. (2010, January 5). Impromptu Interview: Iikka Keranen [Web log post]. Retrieved from http://www.nodraw.net/2010/01/interview-iikka/

Nerurkar, M. (2009, August 25). *No More Wrong Turns*. Retrieved from Gamasutra website: http://www.gamasutra.com/view/feature/4115/no_more_wrong_turns.php

Nolan, C., & Thomas, E. (Producers). (2010). *Inception* [Motion picture]. United States: Warner Bros. Pictures.

Novak, J. (2008). Levels: Creating the World. In *Game Development Essentials* (pp. 208-227). Delmar Cengage Learning.

Peters, C. (2009, August 21). This point ain't gonna cap itself! [Web log post]. Retrieved from TF2 Official Blog: http://teamfortress.com/post.php?id=2754

Riddle, J., & Ayliffe, P. (Speakers). (2009). *Lighting with Purpose* [MP3]. GDC Vault.

Roger, S. (Speaker). (2009). *Everything I Learned About Level Design I Learned from Disneyland* [MP3]. GDC Vault.

Rouse, R., III (Speaker). (2010). *Environmental Narrative: Your World is Your Story* [MP3]. GDC Vault.

Schell, J. (2010). *The Art of Game Design*. Morgan Kaufmann. (Original work published 2008)

Scneider, L. (Speaker). (2010). *Multiplayer Level Design in Red Faction: Guerrilla* [MP3]. GDC Vault.

Silli, E., & EA DICE. (2010, March). *Mirror's Edge: Level Design Challenges & Solutions*. Powerpoint Research Presentation presented at GDC Europe 2010.

Smith, H., & Worch, M. (Speakers). (2010). *"What Happened Here?" – Environmental Storytelling* [MP3]. GDC Vault.

Smith, R. (Speaker). (2009). *Helping Your Players Feel Smart: Puzzles as User Interface* [MP3]. GDC Vault.

Valve Corporation (Speaker). (2007). *Portal Developer Commentary* [Video Game].

Valve Developer Community [Wiki]. (n.d.). Retrieved March, 2011, from Valve Corporation website: http://developer.valvesoftware.com/wiki/Main_Page

Vlachos, A., & Valve Corporation. (2010, July 28). *Water Flow in Portal 2*. Powerpoint Research presentation presented at SIGGRAPH2010.

Appendix A

Glossary

Art Pass: Stage of level development encompassing the transition from a generic prototype to a fleshed-out level indicative of its final aesthetics.

BLU: The team in TF2 that takes the offensive role in payload maps.

Brush: Brushes are the blocks with which the world is created in the Source Hammer Editor.

BSP: The filename for compiled Source maps; **B**inary **S**pace **P**artition

Changelog: The list of changes made in between one iteration and its successor.

Choke/Choking: Chokes, or choke points, are areas designed into a map to force opponents into contact. Choking is the process of bringing in an areas boundary to create a choke.

Clipping: Process of smoothing player flow around world geometry, as well as preventing players from accessing non-gameplay areas.

Cover: Geometry created with the intent to hide players or protect them from incoming attacks.

CP: Capture Point; the name of the ownable objectives in TF2 that must be controlled to win.

Dropdown: A pattern of geometry resembling a hole that represents a one-directionally (downwards) gated pathway.

Flow: The way players move through a map; the patterns of traffic players create as they interact with the map.

Gameplay: General term referring to the condition of a game or level in respect to its game-oriented dynamics. Good gameplay generally leads to a fun experience.

Gate: A pattern of geometry limiting movement to one direction in respect to a player path.

Geometry: The arrangement of brushwork in a level. Generally refers to immobile architecture.

Heatmap: A distribution of events occurring on a map in visual form, laid out over a representation of the map in 2D.

Layout: A term encompassing the entirety of the playable map; also implies reference to the pattern of player flow within that layout.

Optimization: Development process striving to achieve maximum performance of the level on the players' hardware, finding a balance between aesthetics and ease of rendering.

Payload train: The cart in the payload gamemode that must be guided along a track to each CP in succession to win the round as offense.

Playerbase: The entire audience of the game or level; a population of players.

Playtest: Testing process allowing users to experience the game or level as if it were a finished product. This returns data pertinent to actual gameplay.

RED: The name of the team in TF2 that defends on payload maps.

Round: The span of time during which an outcome is decided for a map; in TF2 this lasts from the start of Setup to humiliation time.

Setup: The span of time in TF2 during which offense is restricted to the interior of their spawn, while defense is given leave to prepare for the coming assault.

Source: Valve's current generation game engine, upon which TF2 runs.

Spam/Spammy: Refers to gameplay relatively low in conscious and engaged participants. In reference to TF2, players "spam" when they shoot in an area hoping to hit one another by chance, rather than aiming precisely.

Stop goal: The point on a given metric which, once reached, determines satisfaction with the current iteration and approves the next phase of development.

Spawn wave: The segmented delivery method of dead players back into the game within TF2.

TF2: Team Fortress 2, the game in which the subject of this paper's case study resides.

VMF: Valve Map File, the filename for Source maps before they have been compiled.

VOIP: Voice Over IP, refers to the ability to use one's own real voice in a game as a means of communication.

Appendix B

Formulas

Stop% for CP n:

$$\frac{(\#\text{ ending with CP } n \text{ having been captured last})}{(\text{Total rounds played})}$$

Winning rate for BLU:

$$\frac{(\#\text{ rounds ending with CP4 captured})}{(\text{Total rounds played})}$$

Winning rate for RED:

$$1 - \frac{(\#\text{ rounds ending with CP4 captured})}{(\text{Total rounds played})}$$

Capture Difficulty Peak for CP n:

$$100 \times \frac{(\text{CP } n\ Stop\ \%)}{(1 - (\sum_{i=0}^{n-1} \text{CP } Stop\ \%))}$$

Arbitrary Difficulty Valley before CP n: *(for sake of visualization)*

$$.2 \times \frac{(\text{CP } n\ Difficulty\ + \text{CP } (n-1) Difficulty)}{2}$$

% of Map Experienced on Average per Round:

$$\sum_{i=1}^{4} (\text{CP } n\ Stop\ \%) \times \left(\frac{\text{Segment } n \text{ Length}}{\text{Total Segment Lengths}}\right)$$

% Deviance from Ideal Curve

$$\frac{\sum_{i=1}^{4} \left| \frac{(\text{Actual CP } n\ Difficulty) - (\text{Ideal CP } n\ Difficulty)}{(\text{Ideal CP } n\ Difficulty)} \right|}{4}$$

Appendix C

Data Charts

The following tables contain the Difficulty calculation tables for each respective iteration of the map.

Note: "Point" column refers to the map segment directly after the CP for which it is numbered, with Point 0 equating to the map's outset. As a result, the Fail% and Difficulty for Point 4 have little meaning (as all games where CP4 is captured end exactly on CP4)

Table C1: Iteration 1

Point	Stop %	Cont. %	Fail %	Difficulty
0	50%	100.00%	50.00%	50
1	0%	50.00%	0.00%	0
2	50%	50.00%	100.00%	100
3	0%	0.00%	0.00%	0
4	0%	0.00%	0.00%	0

Table C2: Iteration 2

Point	Stop %	Cont. %	Fail %	Difficulty
0	27%	100.00%	26.67%	27
1	13%	73.33%	18.18%	18
2	0%	60.00%	0.00%	0
3	33%	60.00%	55.56%	56
4	27%	26.67%	100.00%	100

Table C3: Iteration 3

Point	Stop %	Cont. %	Fail %	Difficulty
0	20%	100.00%	20.00%	20
1	50%	80.00%	62.50%	63
2	0%	30.00%	0.00%	0
3	10%	30.00%	33.33%	33
4	20%	20.00%	100.00%	100

Table C4: Iteration 4

Point	Stop %	Cont. %	Fail %	Difficulty
0	30%	100.00%	30.00%	30
1	20%	70.00%	28.57%	29
2	10%	50.00%	20.00%	20
3	20%	40.00%	50.00%	50
4	20%	20.00%	100.00%	100

Table C5: Iteration 5

Point	Stop %	Cont. %	Fail %	Difficulty
0	0.00%	100.00%	0.00%	0
1	25.00%	100.00%	25.00%	25
2	25.00%	75.00%	33.33%	33
3	0.00%	50.00%	0.00%	0
4	50.00%	50.00%	100.00%	100

Table C6: Iteration 6

Point	Stop %	Cont. %	Fail %	Difficulty
0	25.00%	100.00%	25.00%	25
1	0.00%	75.00%	0.00%	0
2	25.00%	75.00%	33.33%	33
3	25.00%	50.00%	50.00%	50
4	25.00%	25.00%	100.00%	100

Table C7: Iteration 7

Point	Stop %	Cont. %	Fail %	Difficulty
0	35.00%	100.00%	35.00%	35
1	0.00%	65.00%	0.00%	0
2	6.00%	65.00%	9.23%	9
3	0.00%	59.00%	0.00%	0
4	59.00%	59.00%	100.00%	100

Table C8: Iteration 8

Point	Stop %	Cont. %	Fail %	Difficulty
0	17.00%	100.00%	17.00%	17
1	0.00%	83.00%	0.00%	0
2	0.00%	83.00%	0.00%	0
3	17.00%	83.00%	20.48%	20
4	67.00%	66.00%	101.52%	102

Table C9: Iteration 9

Point	Stop %	Cont. %	Fail %	Difficulty
0	22.00%	100.00%	22.00%	22
1	4.00%	78.00%	5.13%	5
2	4.00%	74.00%	5.41%	5
3	9.00%	70.00%	12.86%	13
4	61.00%	61.00%	100.00%	100

Table C10: Iteration 10

Point	Stop %	Cont. %	Fail %	Difficulty
0	39.00%	100.00%	39.00%	39
1	0.00%	61.00%	0.00%	0
2	9.00%	61.00%	14.75%	15
3	4.00%	52.00%	7.69%	8
4	48.00%	48.00%	100.00%	100

Table C11: Iteration 11

Point	Stop %	Cont. %	Fail %	Difficulty
0	25.00%	100.00%	25.00%	25
1	8.00%	75.00%	10.67%	11
2	17.00%	67.00%	25.37%	25
3	0.00%	50.00%	0.00%	0
4	50.00%	50.00%	100.00%	100

Table C12: Iteration 12

Point	Stop %	Cont. %	Fail %	Difficulty
0	37%	100.00%	37.02%	37
1	10%	62.98%	15.76%	16
2	5%	53.05%	9.35%	9
3	8%	48.09%	15.87%	16
4	40%	40.46%	100.00%	100

Table C13: Iteration 13

Point	Stop %	Cont. %	Fail %	Difficulty
0	28%	100.00%	27.66%	28
1	9%	72.34%	11.76%	12
2	9%	63.83%	13.33%	13
3	4%	55.32%	7.69%	8
4	51%	51.06%	100.00%	100

Table C14: Iteration 14

Point	Stop %	Cont. %	Fail %	Difficulty
0	31%	100.00%	31.11%	31
1	13%	68.89%	18.28%	18
2	10%	56.30%	17.11%	17
3	6%	46.67%	12.70%	13
4	41%	40.74%	100.00%	100

Table C15: Iteration 15

Point	Stop %	Cont. %	Fail %	Difficulty
0	14%	100.00%	14.20%	14
1	20%	85.80%	23.80%	24
2	10%	65.38%	16.00%	16
3	10%	54.92%	17.91%	18
4	45%	45.08%	100.00%	100

Table C16: Iteration 16

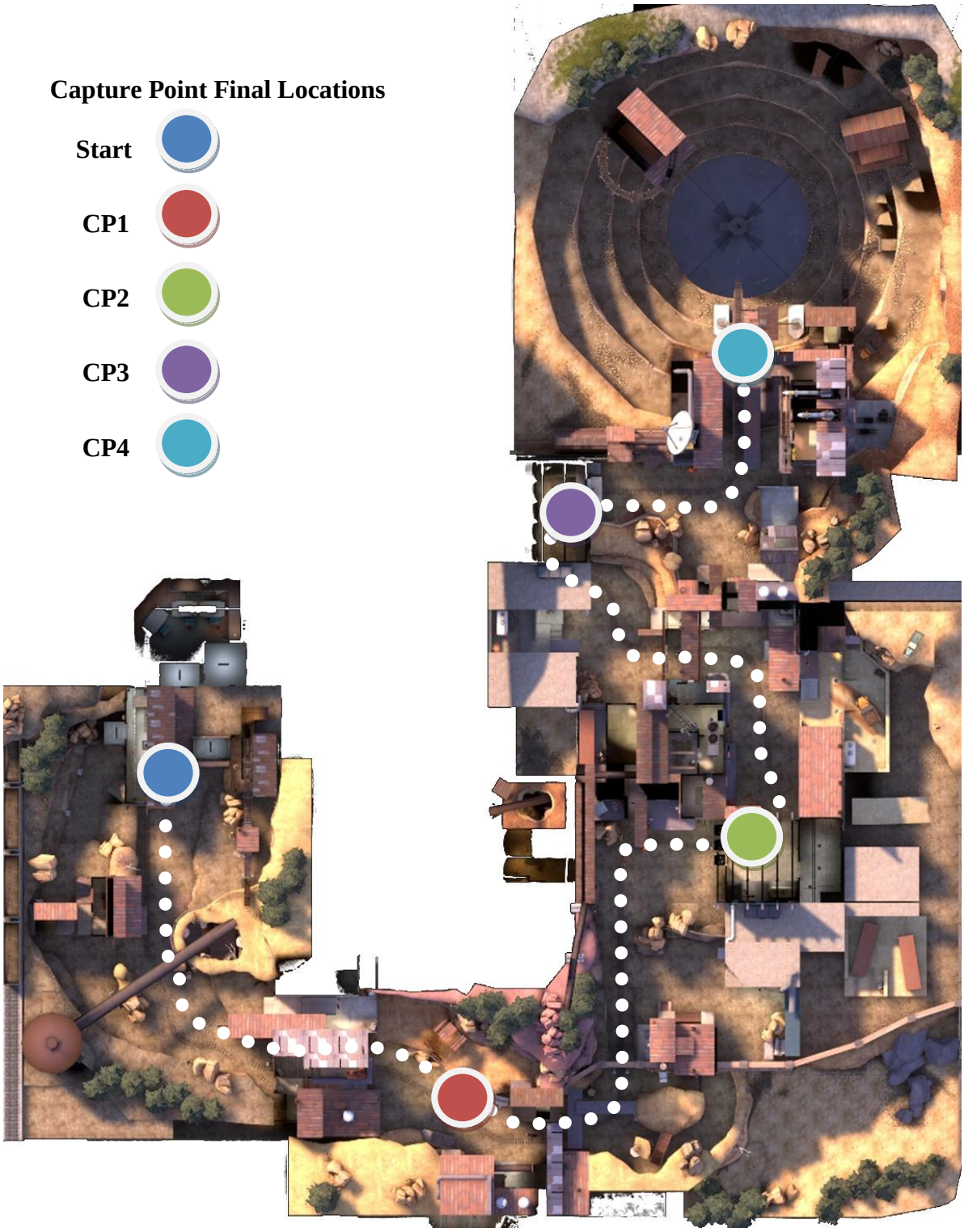
Point	Stop %	Cont. %	Fail %	Difficulty
0	14%	100.00%	13.91%	14
1	18%	86.09%	20.82%	21
2	12%	68.17%	17.67%	18
3	11%	56.13%	19.28%	19
4	45%	45.30%	100.00%	100

Table C17: Iteration 17

Point	Stop %	Cont. %	Fail %	Difficulty
0	14%	100.00%	13.91%	14
1	14%	86.09%	16.26%	16
2	15%	72.09%	20.81%	21
3	15%	57.09%	26.27%	26
4	45%	42.09%	107.63%	108

Appendix D

CP Location Reference



Appendix E

Payload Cart Evolution

Figure 0:1 Original Artist Proposal

Figure 2 Silhouetted model guideline

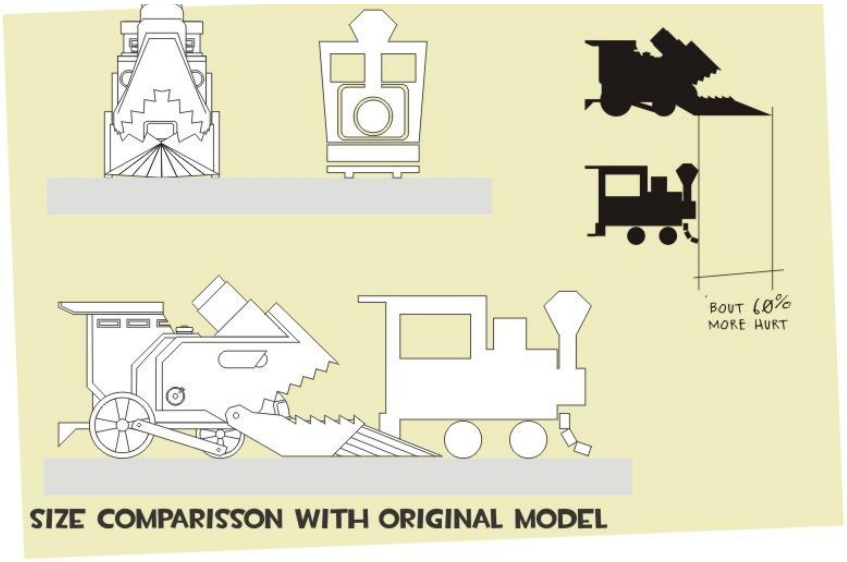


Figure 3 Level designer feedback

Figure 4 Revised concept

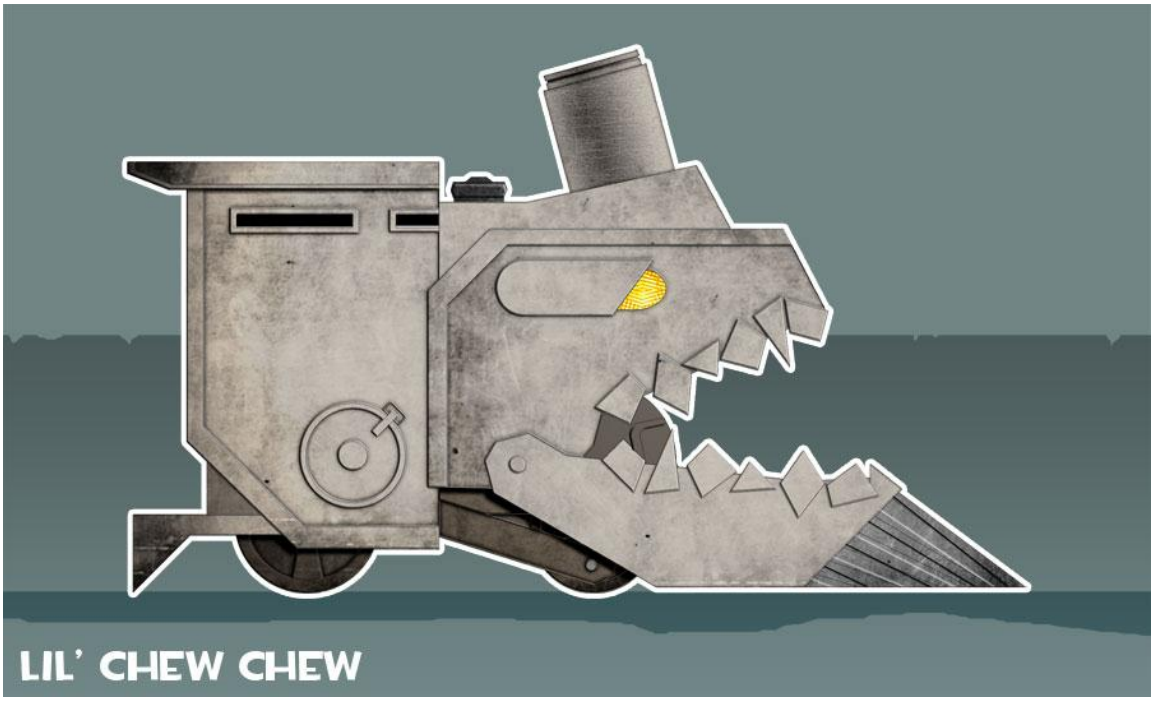
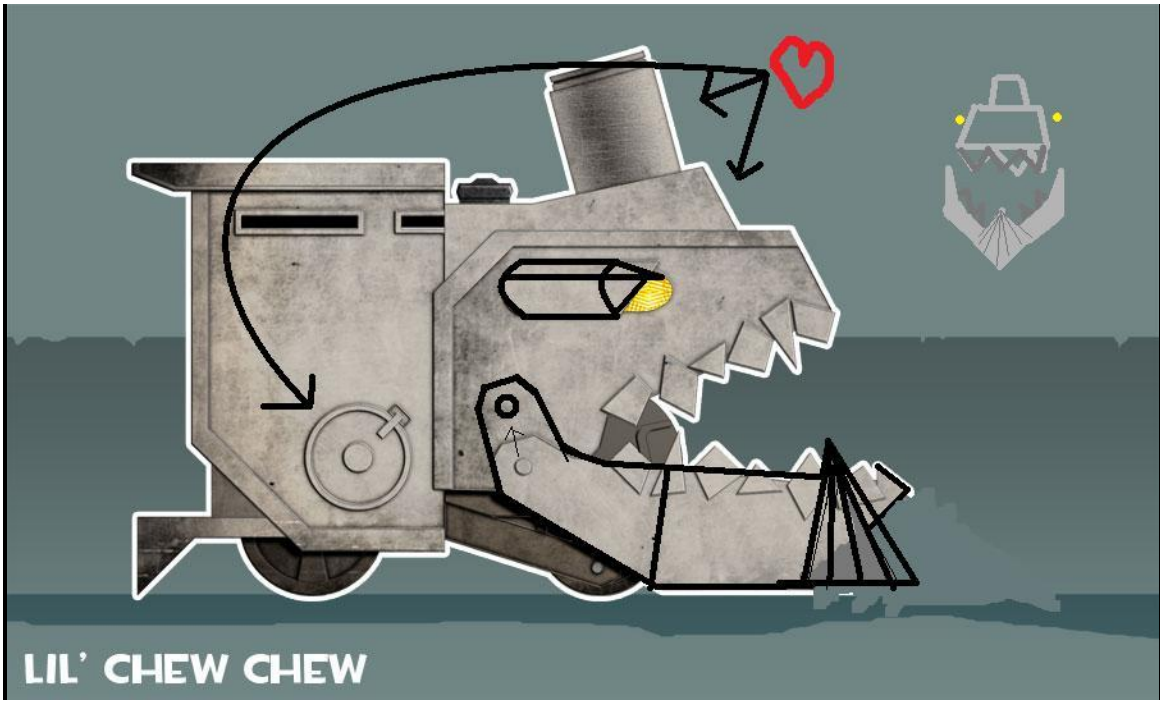


Figure 5 Level designer feedback



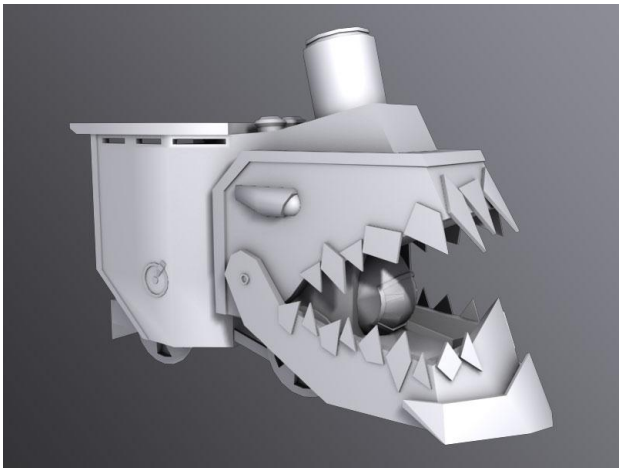


Figure 6 First model, pre-unwrapping

Figure 7 Paintover of untextured model

Figure 8 Fully textured model

Figure 9 Textured model, Beta 1

Figure 10 Finished model, Beta 2

VITA

Patrick Mulholland

10727 Alexander Mills
Charlotte, NC 28277
patrick.f.mulholland@gmail.com (704-608-4025)

EDUCATION:

The Schreyer Honors College at the Pennsylvania State University
The Pennsylvania State University, University Park, PA, 5/2011
B.S. Information Sciences and Technology
Honors in Information Sciences and Technology

Meikai University, Tokyo, Japan 3/2010 – 7/2010
IES Study Abroad Language Intensive Program

EXPERIENCE:

Duke Energy, Charlotte, NC **[Summer 2009]**
LIT Metro Team

- Addressed CSC client support tickets
- Updated and managed inventory at multiple sites
- Worked extensively in Asset Center databases
- Performed inventory networking and installation tasks

Newsroom Solutions, Inc., Altoona, PA **[Summer 2008]**
Support Technician

- Cooperated directly with clients to interface relevant databases with their news-tickers
- Wrote code in a derivative of PERL to accomplish this
- Served client tickets across multiple time zones

The Pennsylvania State University, University Park, PA **[Spring 2008]**
Research Assistant

- Worked with a professor regarding research in distributed team acclimation methods
- Presented findings in an oral presentation at Penn State's Undergraduate Research Fair
- Analyzed data provided from student participants regarding teamwork strategies

Level Design
Part-time Enthusiast

- Independent study spanning 2007-2011
- Content downloaded and played by tens of thousands of players
- Content published by Valve Software in the PC game "Team Fortress 2"
  - Multiplayer map "Yukon" published August 2009
  - Multiplayer map "Frontier" published February 2011

Accenture, Philadelphia, PA **[Anticipated Summer 2011]**
Entry Level Consulting Analyst

AWARDS & HONORS:

Eagle Scout received badge in 2007
University Dean's List recipient