Symbiosis

W. Andrew Hair

BA (Hons) Digital Music and Sound Arts 3

Student Number: 12824882

Theory 5 (AG317a)

Final Project Documentation

1st May 2015

Contents

Work Synopsis

Symbiosis is an audio exploration game which places the player within an abstract, symbolic ecosystem which responds to each player action with music. *Symbiosis* was developed with the intent of exploring ideas of musical environments, ecology in games, and reciprocal relationships between species. Each interaction in the game presents the player with an opportunity for self-preservation, or altruism. This was achieved by creating a mock ecosystem, complete with plant-life and animals, each of which reside in a symbiotic relationship with other members of the ecosystem, including the player themselves. The relative states of these populous species are mapped to elements of a generative music system. As such, each action the player takes has a direct, or indirect influence upon the music and soundscape. *Symbiosis* was developed using the Unity game engine, Wwise audio engine, and scripted in C#.

Artist Statement

Symbiosis is a stunningly presented musical exploration game in which the player is situated as an autonomous component of an evolving, diverse ecosystem. Players are invited to interact with the ecosystem as they desire, with the impacts of these actions being conveyed through a beautifully realised soundscape which rewards the player for altruistic choices, and scorns them for self-indulgence.

Development of this ecosystem involved the creation of several interacting agents, which take the form of plants and animals of various species. These agents have varying degrees of interdependence on other species. Birds, for example, are dependant on the fruits produced by plant-life, and as such any change in the prevalence of fruit-bearing plants have an indirect impact on the bird population. These impacts are conveyed through change and evolution in the musical soundscape. Inspiration for this was found in games such as *Fract OSC*, which places the player within a 'musical environment', in which the various architecture and lifeforms could be considered individual components within an immense synthesizer. *Fract OSC* then tasks the player with solving puzzles that involve creating harmony and structural coherence between these numerous individual components. Additionally, *Proteus* is an exploration game which presents the player with a similar musical environment, constantly evolving based on season, weather and location. Each organism in the environment also has a unique musical signature, which come together as a whole to produce beautiful and diverse soundscapes. *Proteus*, however, involves little player interaction other than in the autonomous exploration of its environment.

With *Symbiosis,* I intended to fuse these ideas of evolving musical environments, with musical player interactions. As such, the fundamental actions the player may perform, such as planting trees, or catching fish, are inherently musical. This often involves the player performing a sequence of actions in time with music. The intention behind this is to convey the idea to the player that the game world is governed primarily by music, and that the player can act as both passive receiver, and reactive agent to the soundscape. When attempting to use music as a gameplay mechanic, rhythm and meter are particularly useful as they provide discrete timings which can easily be used to 'challenge' the player, by asking them to perform actions in synchrony.

As a game concerning ecosystems and our influences upon them, *Symbiosis* touches upon the topic and ongoing debate surrounding sustainable culture in our own world. However, my intention here is not to espouse a personal perspective on these issues. Instead, it is to present to the audience a constructed scenario, which stays as true as possible to scientific knowledge of ecosystems, where our actions and their consequential effects are more readily apparent, and represented as the constant changes in musical composition. *Symbiosis* poses the question, through the conduit of music; if the resultant effects of our actions were quicker in coming full circle, how would we then choose to act?

In a similar vein, *Symbiosis* draws upon work in the field of acoustic ecology and environmental soundscapes, primarily that of R. Murray Schafer and the World Soundscape Project. Topics of particular relevance are those of soundscape ecology, which studies the degradation and extinction of naturally occurring sounds, and how these sounds could be preserved. *Symbiosis* explores these topics by presenting the player with choices as to how they wish to interact with the audible ecosystem, where focusing primarily on self-preservation carries the consequence of creating a more monotonous, sparsely populated soundscape. The reverse of which is also true, where a focus on ecological preservation will maintain a more diverse soundscape, at the expense of personal favour. *Symbiosis* also postulates as to the potential for conscious listening to a natural soundscape as a method of understanding the relative health of that ecosystem and its inhabitants. Just as it is common practice to measure the population of a bird species by collating visual sightings, is it not equally as possible to listen with attention to their vocalizations to achieve the same?

In recent years, computer games have garnered extensive recognition as a medium in which artistic ideas can take root and create profound experiences for the audience. Whilst a significant portion of contemporary games still aspire to the artistic ideals of other media, such as film and television, a growing number of developers are looking to explore the potential of the interactive audio-visual form, game or not. In many ways, the advent and growth of these 'art games' resembles the development of interactive and installation art. The audience is encouraged to interact with the work, and how meaning is derived from these experiences is a result of how they interact. *Symbiosis* is most likely to fall within this field of art games, and more broadly with art in general, within the field of interactive, constructed spaces, as well as sharing many ideas with that of sound art.

Audio Implementation

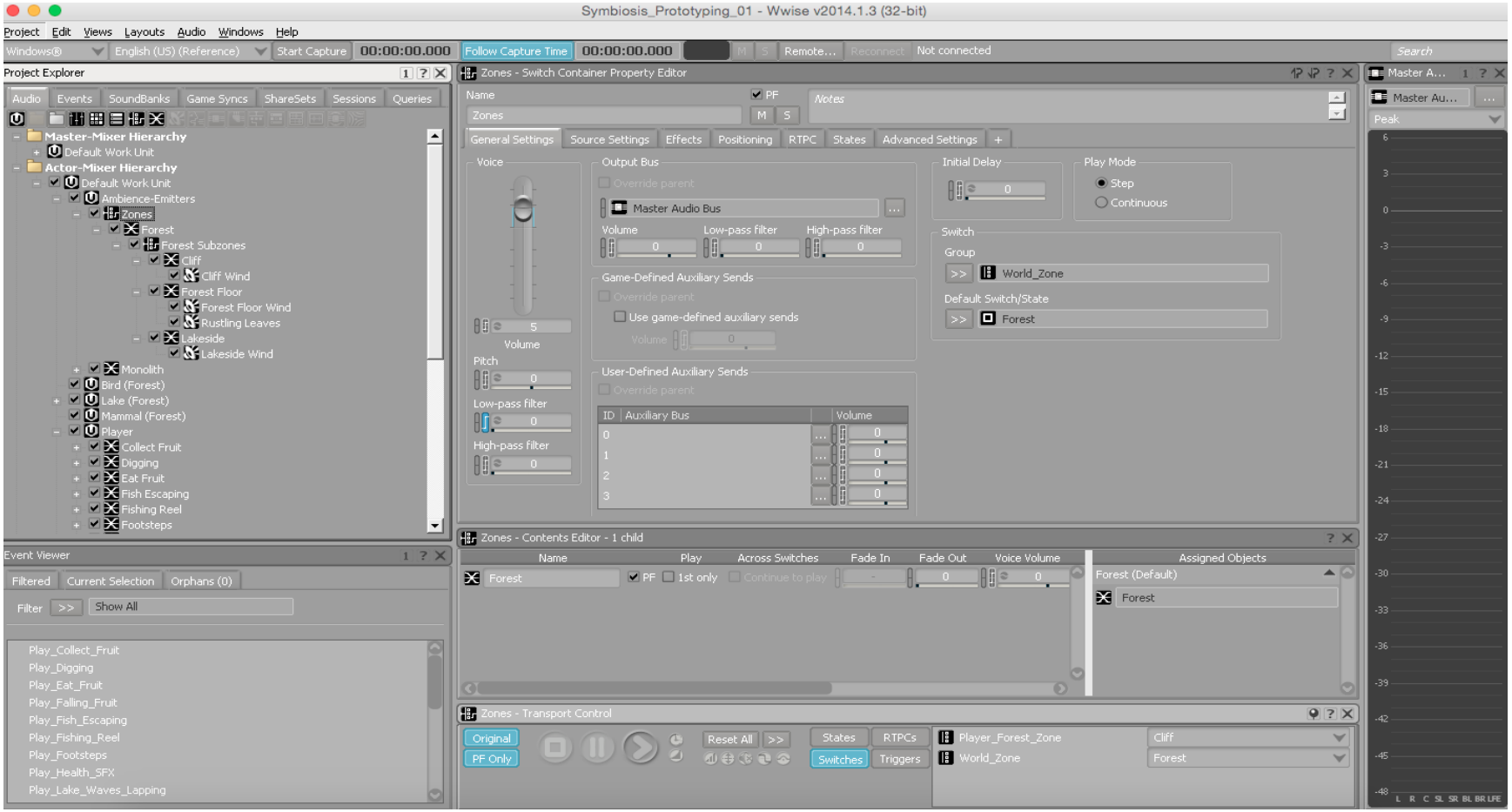
For the sound engine, I used an 'audio middleware' program called Wwise. This software is gradually becoming the industry standard for audio implementation in games. It opens up a wider range of tools and techniques than would be possible in any other sound engine, and essentially acts as a plug-in for the Unity engine. It also gives greater possibilities for audio to influence the game engine, and vice versa. For example I was able to send tempo information from the Wwise music system, into game mechanic scripts in Unity. This was then used to create challenges for the player that involve keeping in time with the music, something that would have been extremely difficult with Unity's built-in audio engine. I will go into great detail on how I developed the audio implementation below.

Ambience /Zone Implementation and Sound Design

The game world is divided into distinct 'zones' using large trigger colliders in the Unity engine, which is covered in the game development segment of this documentation. These three zones - the forest floor, the lake area, and cliffside, each have their own ambient soundscape, including winds, rustling tree leaves, lapping lake water, and so on. When the player moves between each of these zones, the relevant soundscapes fade in/out to create a natural sounding transition between areas. Where visual cues may be lacking in terms of portraying each of these areas (the cliff doesn't so much look like a cliff in the game, due to the abstract nature of the graphics), I tried to exploit sound to its full potential in portraying the detail and character of these zones.

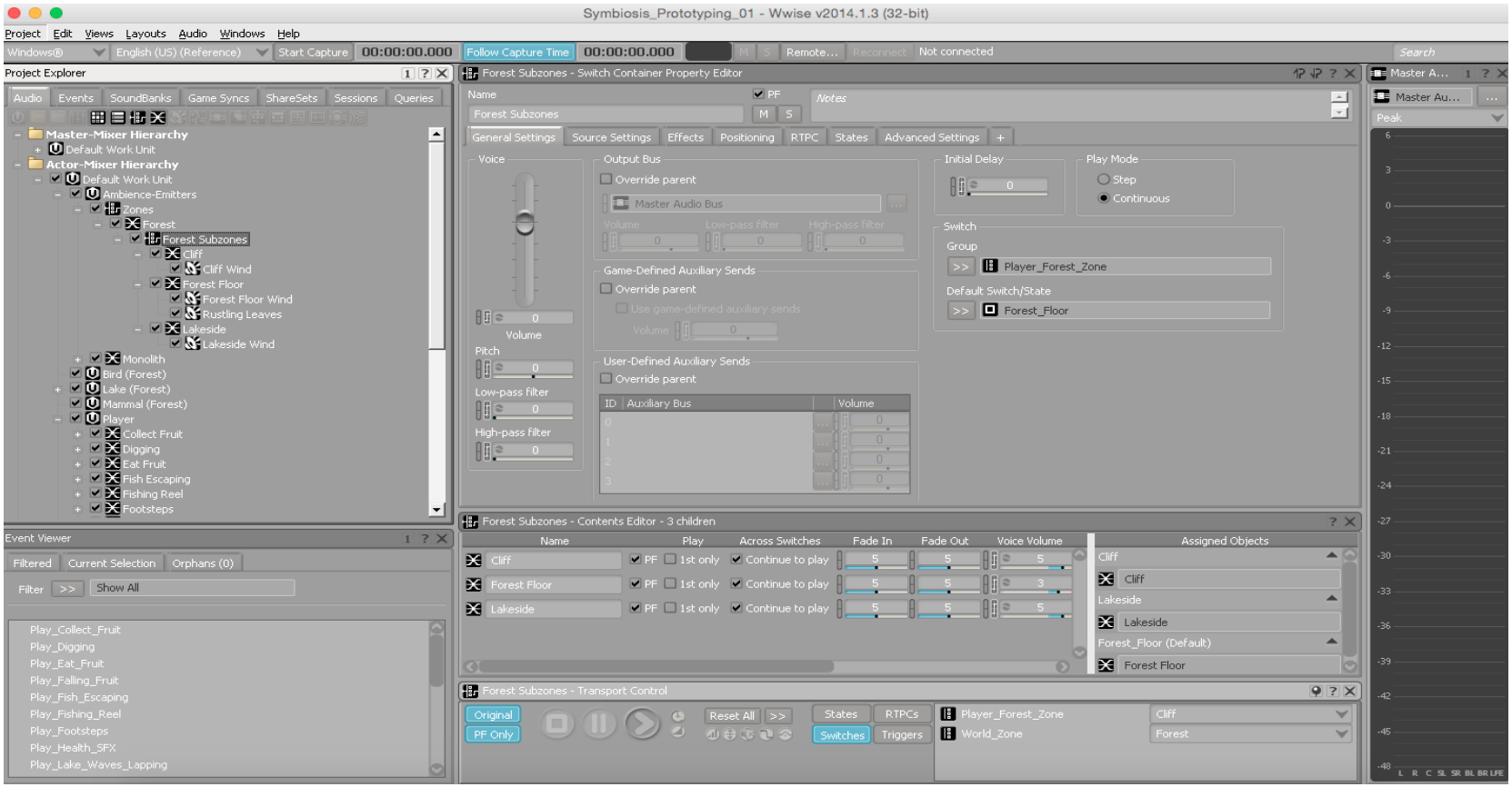
World Zone Switch Container

The 'Zones' object highlighted in the Project Explorer on the left hand side panel, is a Switch object that takes information from the game engine, as to which part of the world the player is in at any given time, and uses this to transition between sets of sounds assigned to each of these zones. In this case, the player is always in the Forest area. As the project grows and includes more levels (mountain ranges, deserts, etc), this would expand.



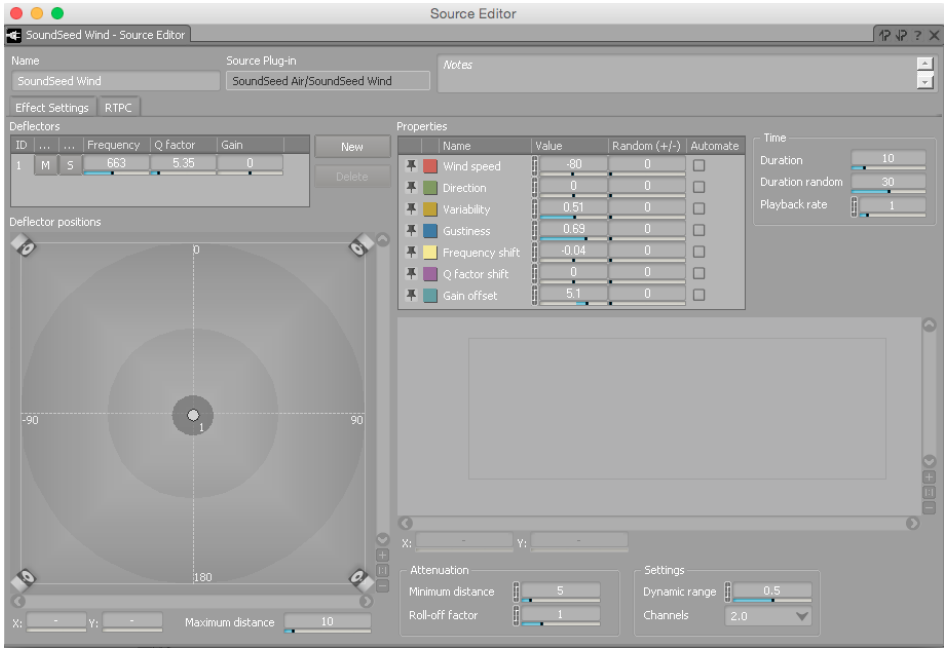
Forest Subzone Switch Container

The 'Forest Subzones' container in the left hand side project explorer functions in essentially the same fashion as the 'World Zone' container listed above. In this case, the container is switching between the three sub-zones within the 'Forest' game level. Namely, between the Forest_Floor, Lakeside, and Cliff sub-zones. The sound elements within this container are set to play continuously, even after transitioning. This ensures there are no sudden switches when moving from one zone to the other, instead they crossfade smoothly.

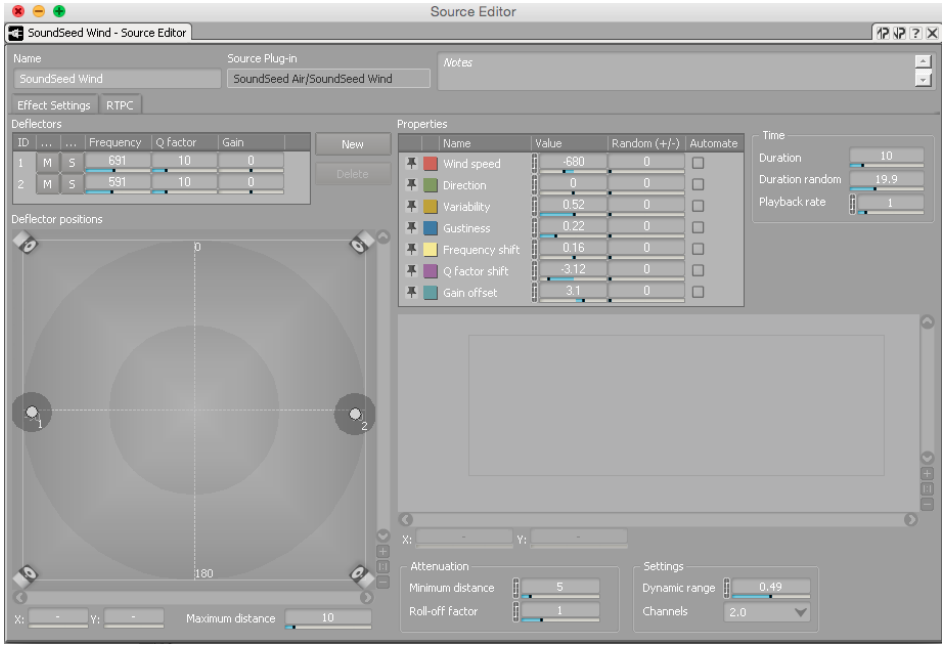


Generative Wind Patches

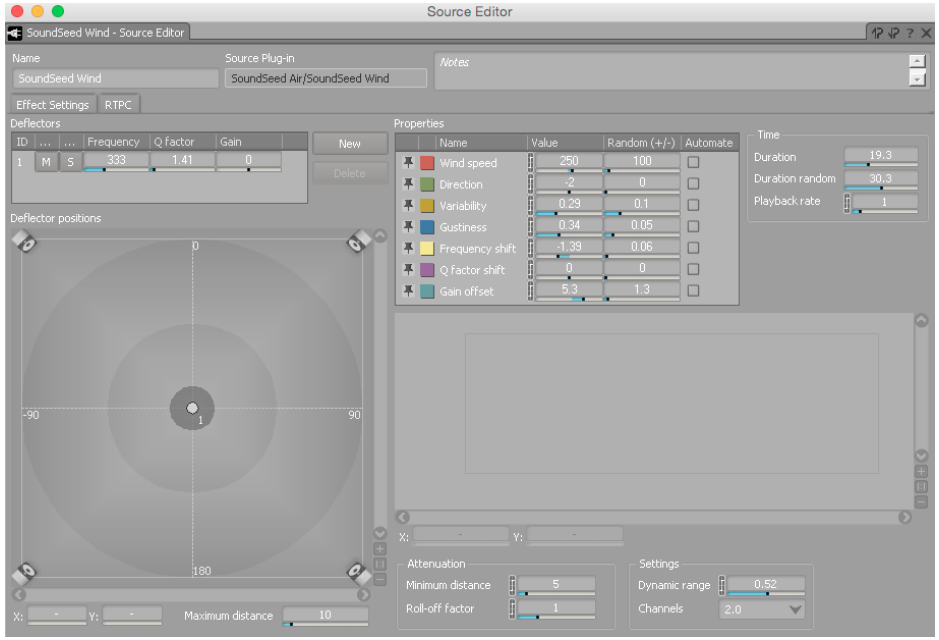
Within the forest subzone containers (Forest Floor, Lakeside, Cliff) are generative wind noise patches which simulate wind characteristic of these kinds of areas. The Cliff patch generates blustery, howling winds that would be characteristic of standing atop a mountain cliff. The Forest Floor patch creates steady, calm wind, emphasising lower frequencies. The Rustling Leaves patch which plays alongside the Forest Floor patch, creates noise akin to wind blowing through the forest canopy. The Lakeside wind is slightly gusty, but high in frequency, as if one was standing on the edge of a still lake with a breeze in the air.



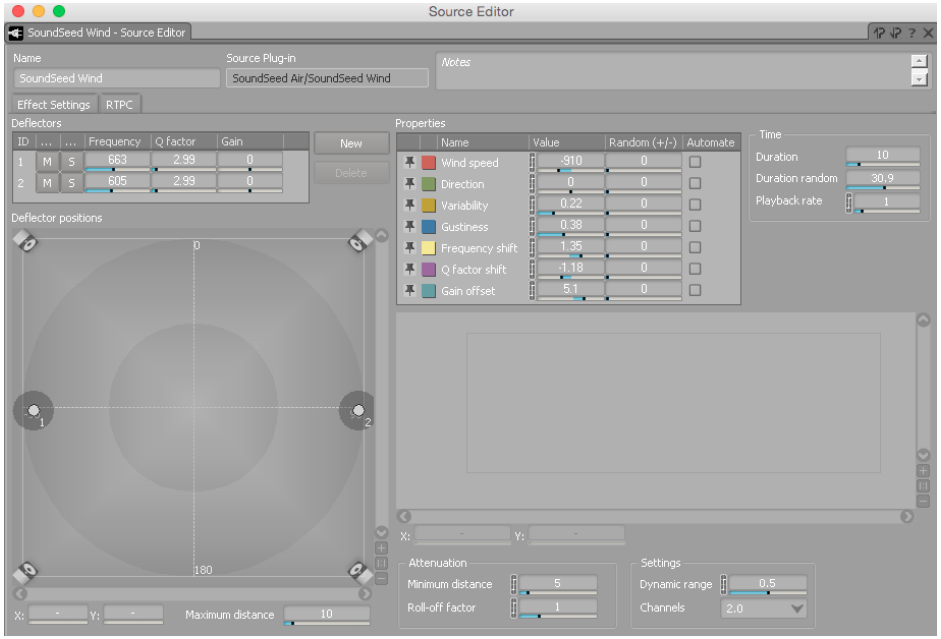
Cliff patch



Forest Floor wind patch



Rustling Leaves wind patch



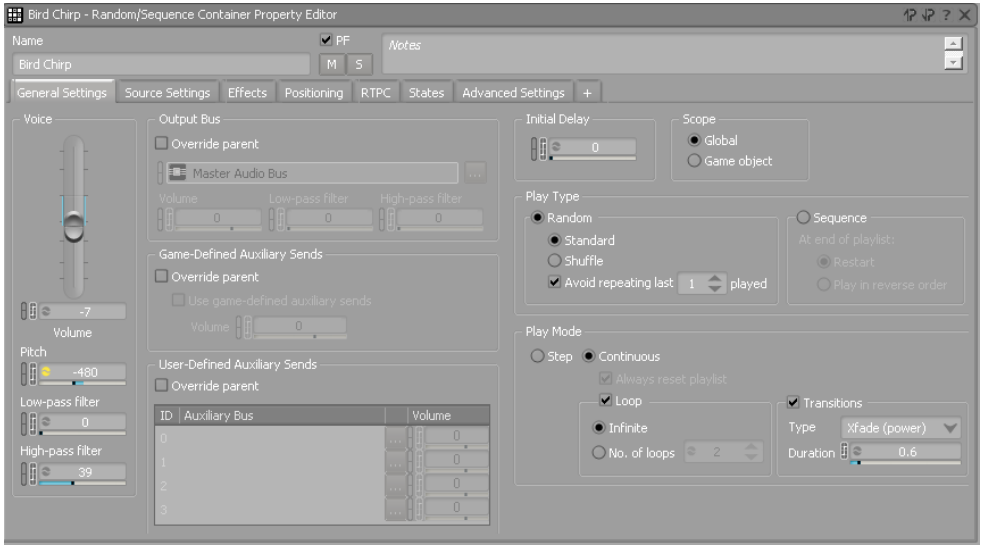
Lakeside wind patch

Game Object Emitters

Alongside the ambient winds and noises, which play continuously in non-diegetic space, are various sound emitters which are assigned to objects within the game world. These emitters are positioned to match the location of the game object. For example, each tree in the world creates its own emitter, which plays sounds of bird song from within the branches. These sounds are also attenuated, meaning as the player moves away from the sound source, the sounds decrease in volume, and higher frequencies are EQ'd down. As follows are a set of examples showing various emitter objects that are localised in the game world.

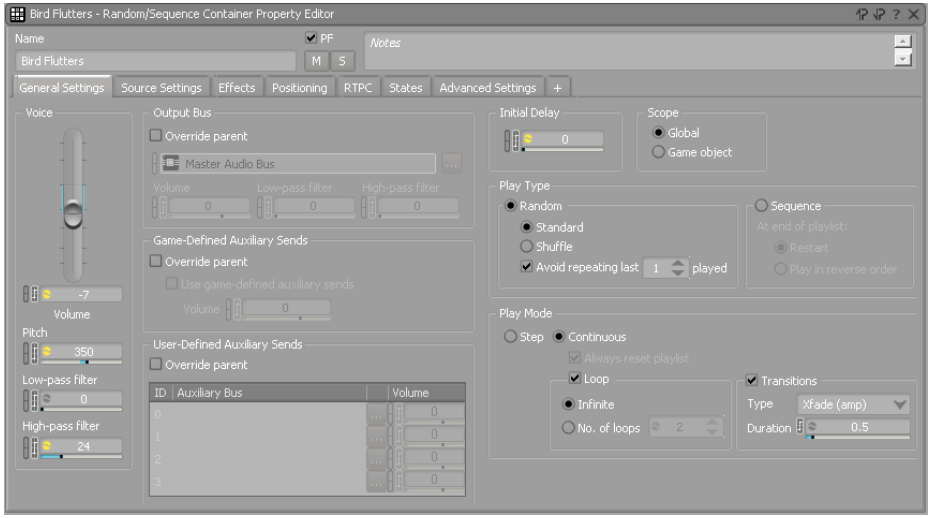
Bird Song

One element of the tree emitters is the sound of bird song emanating from within the tree branches. This was created by setting up a 'Random Container', which randomly selected short samples of synthesized bird song, interspersed with periods of silence. The pitch at which these samples play back was also randomized, within a set range, to create a wide variety of bird sounds.



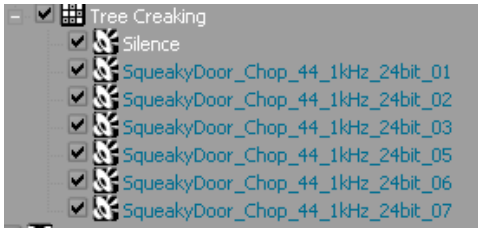
Bird Flutters

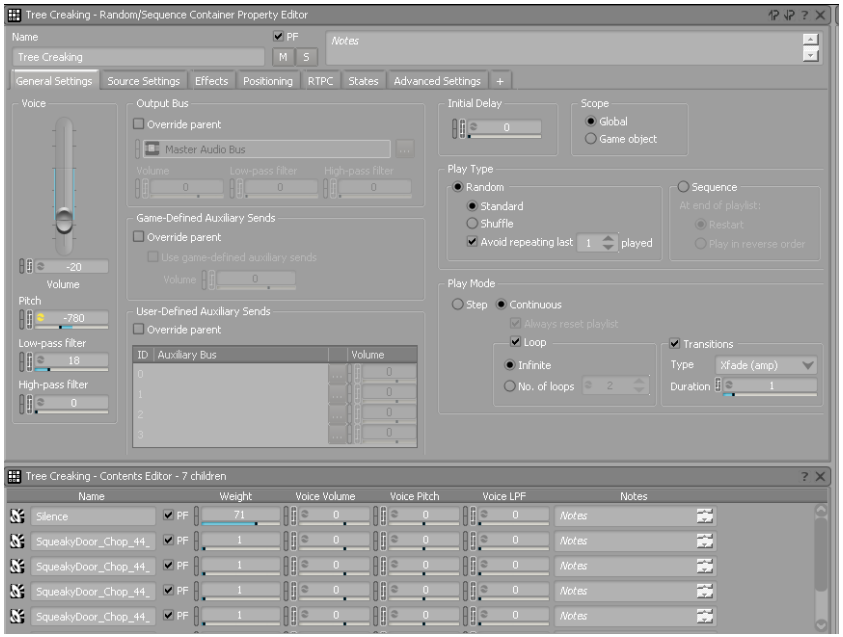
As with the bird song, I added foley samples of birds flapping as they take off from the trees. This was also set to play on a random basis. The container predominantly plays silent, but once in a while it will play a flapping sound in a nearby tree.



Tree Creaks

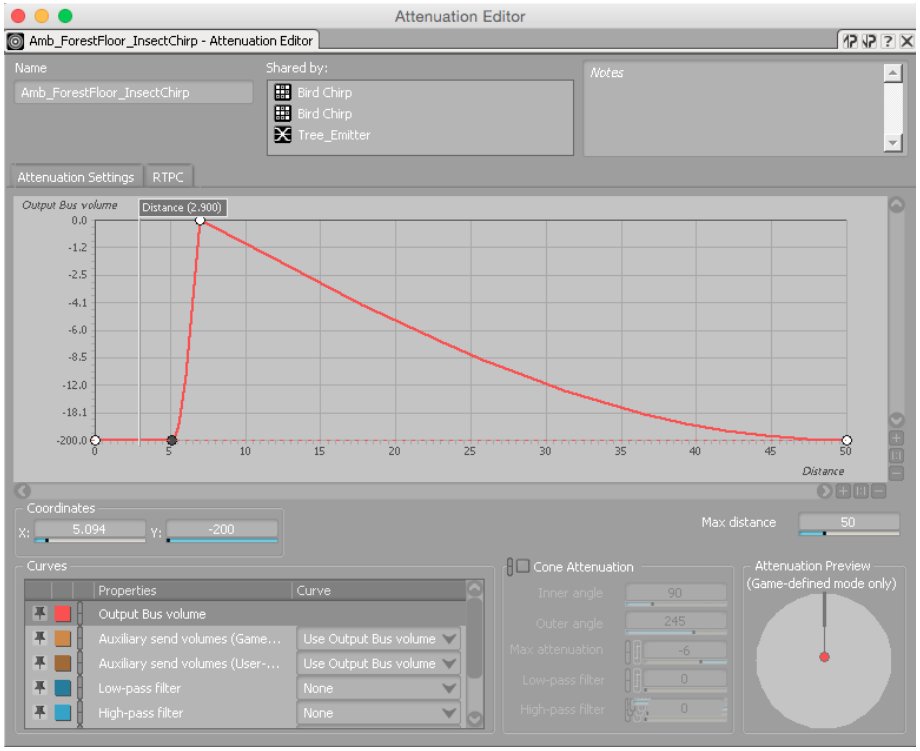
Similarly, the trees will occasionally emit the sounds of branches creaking gently in the wind. This was done to help add a sense of depth and natural variance to the forest soundscape.

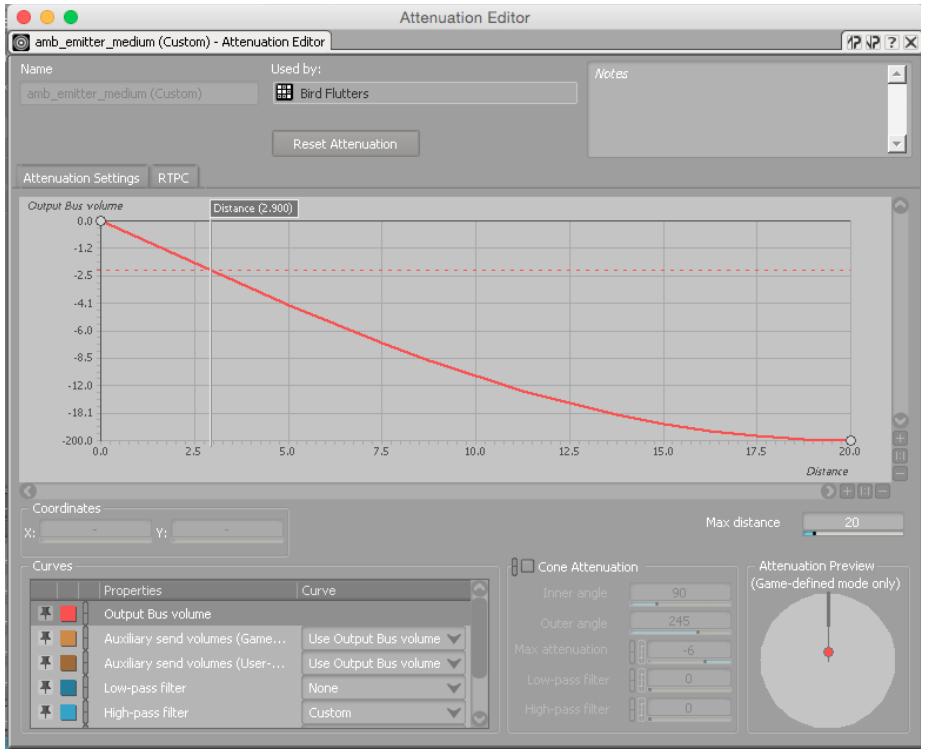




Tree Emitter Attenuations

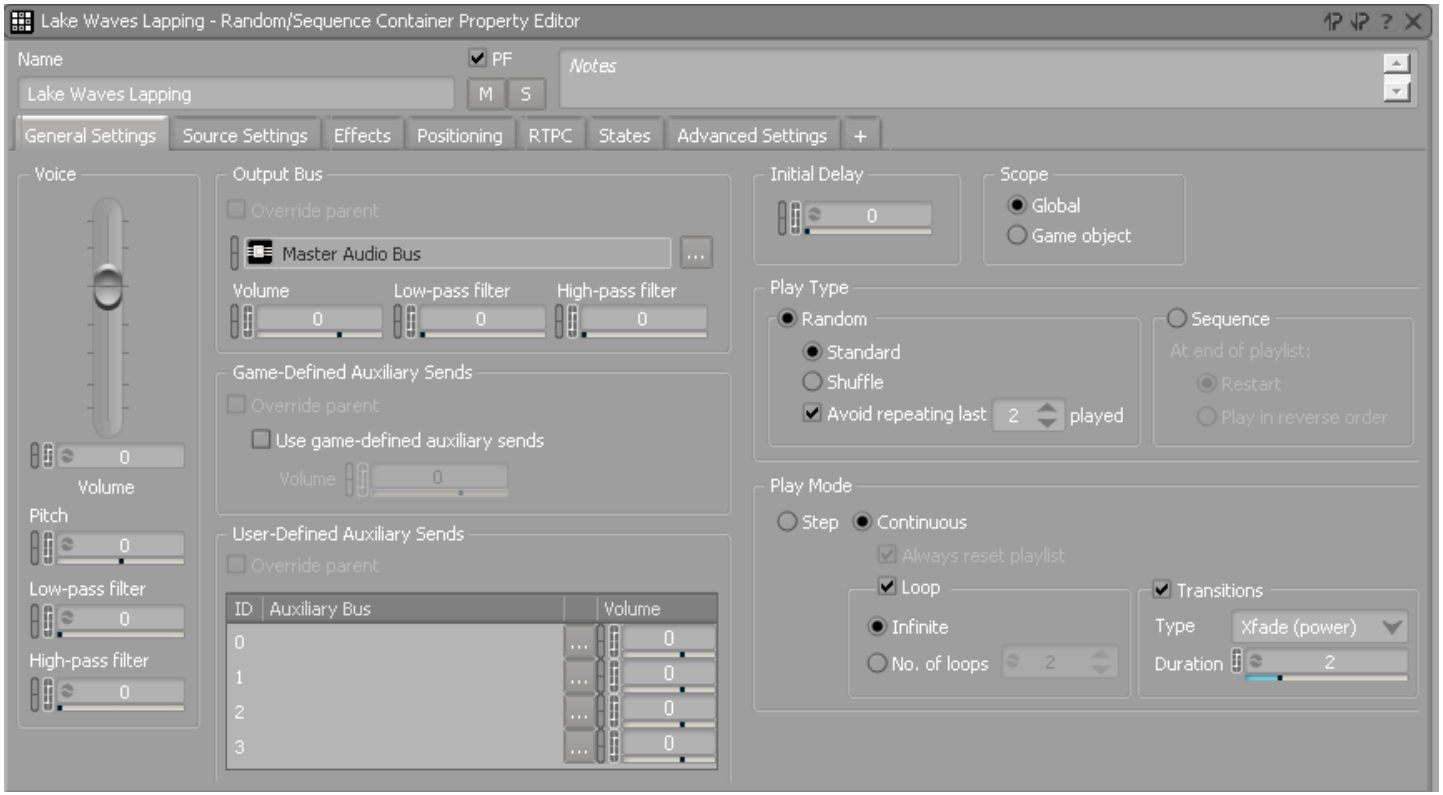
The sounds sourced from tree objects are attenuated so that as the player moves away from the object, the sounds will decrease in volume and frequency range. In addition, if the player moves very close to the tree, the sound of the birds chirping will die away completely, as if the birds had been scared away by the presence of the player (shown in the image on the left).





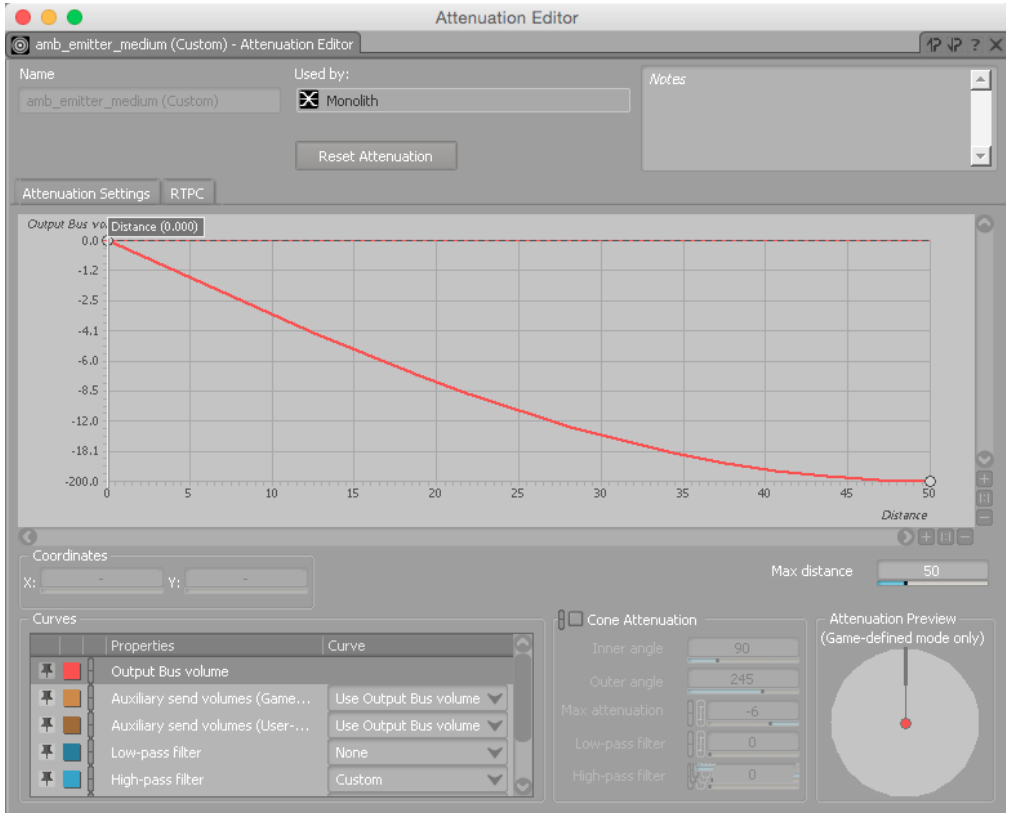
Lake Water Splashes

Around the edges of the lake in the game world, I added many small emitters that play sounds of water gently lapping at the shore. This was created by setting up a 'Random Container' which continuously loops short samples of water splashes/lapping, selected at random. These point emitters are attenuated so that they can only be heard when the player is in close proximity to the lake shore.



Monoliths

The Monolith objects situated in the various subzones in the game world, of which the player can interact with and be presented with short segments of narrative script and instructions on how top play the game, each emit a continuous, evolving drone. This was designed so that these monoliths can be heard from some distance, and the sounds that they emit are distinct from those of the music/ambience. This is so that the player may be drawn to these objects when they are first introduced to the world, and as such they draw attention to themselves.



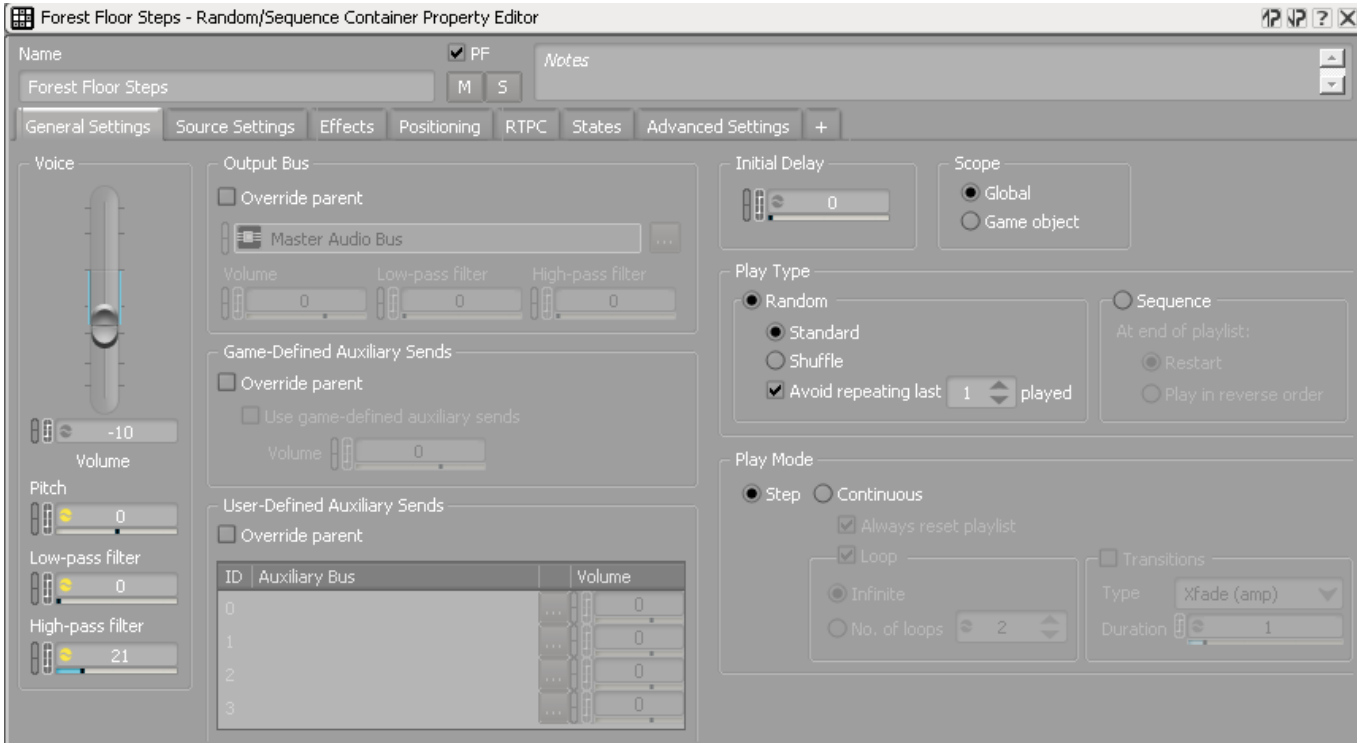
Player Interactions

Each interaction the player may engage in has been designed so that they can deliver positive or negative feedback, in the form of sound design. In many ways, I wanted to describe these interactions with such detail, that the game could function well even without a visual element. That each interaction could generate so descriptive, that the players choices and interactions could create a diverse and detailed audio narrative. In addition, the complexity of the interactions and their meanings that I am trying to convey, is such that developing a complete visual image (through animations, 3D models, characters, etc) is far beyond my skill set. Developing game worlds with such a high degree of detail or realism, is usually the domain of big-budget studios with years of development time to work with. Instead, I tried to exploit sound design to communicate the meaning of these interactions as much as possible.

Footsteps

Symbiosis places the player in a game world which is experienced by controlling a 'First-Person' character. This means that they experience the world almost literally through the eyes of the character. As such, the sounds we experience in our own first person experience of the real world, must be transposed or conveyed into the game world, in order to add coherence to the virtual first-person perspective. Footsteps are a fundamental component of this perspective, and great care was taken to create believable footstep sounds.

The 'Surface Type' switch container chooses which footstep sample set to play depending on where the player is located in the game world. Surfaces in the game world are assigned to 'Rock' or 'Forest Floor', and the container will switch between these sample sets when the player moves across these surfaces.

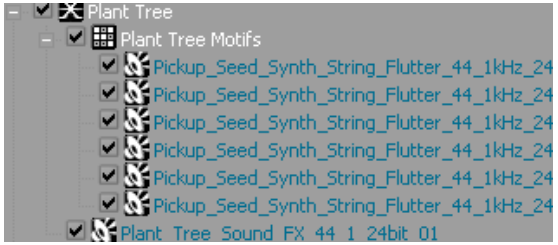


The individual footstep foley samples themselves are contained within a 'Random Container' which selects individual footstep samples at random. When these samples play back, their pitch and filtering are randomized (within a set range). This is to help ensure the player does not encounter 'ear fatigue', where the player essentially recognizes that the sounds of the footsteps are repeating themselves, and therefore are digital samples, thus breaking a sense of immersion in the game world. When a footstep sample is set to play is determined by a Unity script which sends events to Wwise that recur at set intervals depending on the movement speed of the player character.

```
// Use this for initialization
void Start () {
    controllerFinder = this.GetComponent<CharacterController> ();
}

// Update is called once per frame
void Update () {

            if (controllerFinder.velocity.magnitude < 0.5) { //if the player is slowing down, cancel the footstep repeater
                    stillMoving = false;
            }
            if (controllerFinder.velocity.magnitude > 0.5) {// if the player is speeding up, start the footstep repeater
                    if (!routineRunning) {
                            routineRunning = true;
                            StartCoroutine (FootstepTiming ());
                    }
                    stillMoving = true;
            }

            if (controllerFinder.velocity.magnitude > 4) {
                    footstepInterval = 0.45f;
            } else if (controllerFinder.velocity.magnitude > 3 && controllerFinder.velocity.magnitude < 4) {
                    footstepInterval = 0.5f;
            } else if (controllerFinder.velocity.magnitude > 2 && controllerFinder.velocity.magnitude < 3) {
                    footstepInterval = 0.6f;
            } else if (controllerFinder.velocity.magnitude > 1 && controllerFinder.velocity.magnitude < 2) {
                    footstepInterval = 0.75f;
            }
}

IEnumerator FootstepTiming (){

    while (stillMoving) {

        AkSoundEngine.PostEvent ("Play_Footsteps", this.gameObject);

        yield return new WaitForSeconds (footstepInterval);
    }

    routineRunning = false;
}
```

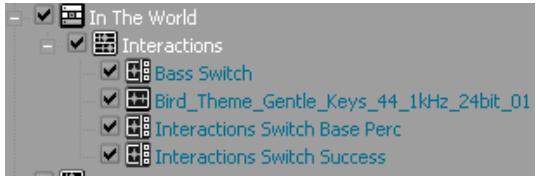
Collecting, Eating, Feeding and Planting

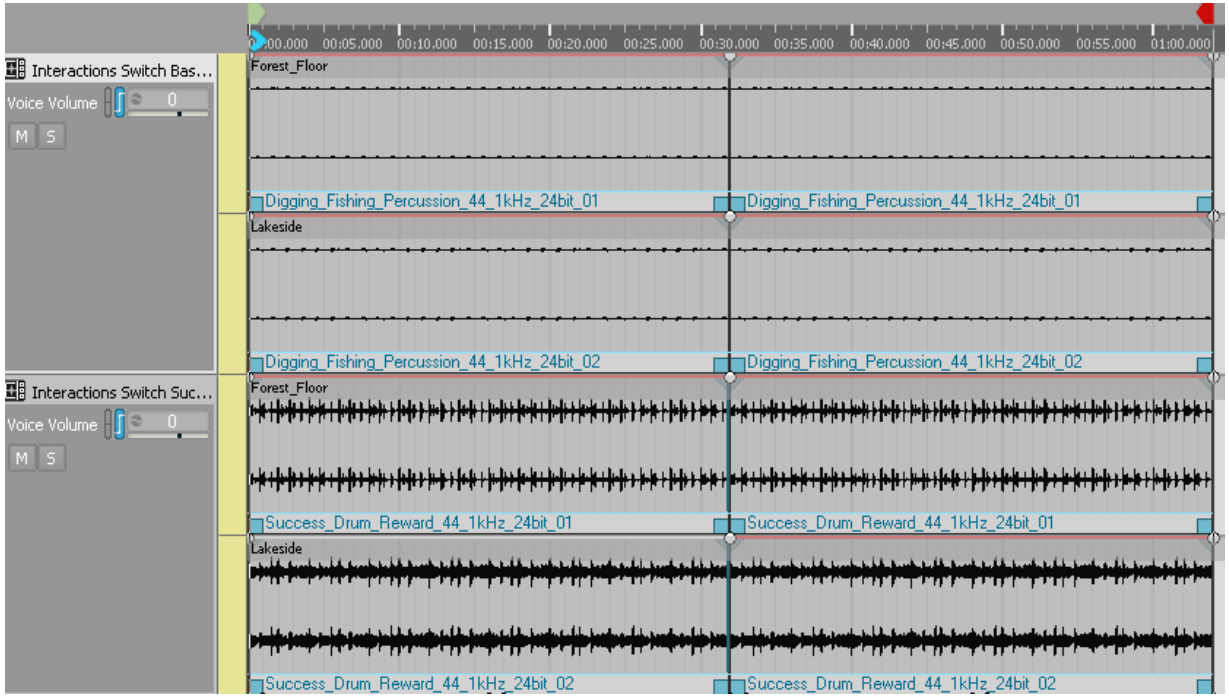
Some small interactions also have short sound cues, such as collecting fruits, eating a fruit or fish, planting a tree, and feeding the fish in the lake. These sounds were designed to reinforce or confirm that the player has successfully interacted, and carry semantic connotations as to the action they have performed. For example the sound of planting a tree is a heavy thud, followed by the stretching and creaking of branches, as if the tree had just grown a large amount in a short space of time.



Digging and Tree Planting Interaction Sequence

Some interactions in the game require the player to hit a sequence of controller buttons in time to music. This was done by sending tempo information from the Wwise music engine, and having this control when a player can successfully hit the controller buttons.





The Wwise music engine allows for synchronized transitions between musical segments. It can also send messages to Unity at user-defined cue points, on beats, bars, phrases or grid endings. To keep the interaction sequence musically in time, I created a script which receives a message at the next bar, which initiates the button input sequence. I then play a 'tick' sound for the next three beats, much like a 'count in' or metronome. This allows the player to get comfortable with the timings. After these three introductory beats, the player is presented with direction arrows on screen that should be pressed in time with the following beats.

```
IEnumerator DigSequence (){

    yield return new WaitForSeconds (initialDelay);

    while (!barThisFrameRef)
    {yield return null;}

    AkSoundEngine.PostEvent("Play_testbeep2", this.gameObject);

    yield return new WaitForSeconds (0.1f);

    while (!beatThisFrameRef)
    {yield return null;}

    AkSoundEngine.PostEvent("Play_testbeep2", this.gameObject);

    yield return new WaitForSeconds (0.1f);

    while (!beatThisFrameRef)
    {yield return null;}

    AkSoundEngine.PostEvent("Play_testbeep2", this.gameObject);

    yield return new WaitForSeconds (0.1f);

    while (!beatThisFrameRef)
    {yield return null;}

    AkSoundEngine.PostEvent("Play_testbeep2", this.gameObject);

    yield return new WaitForSeconds (0.1f);
```

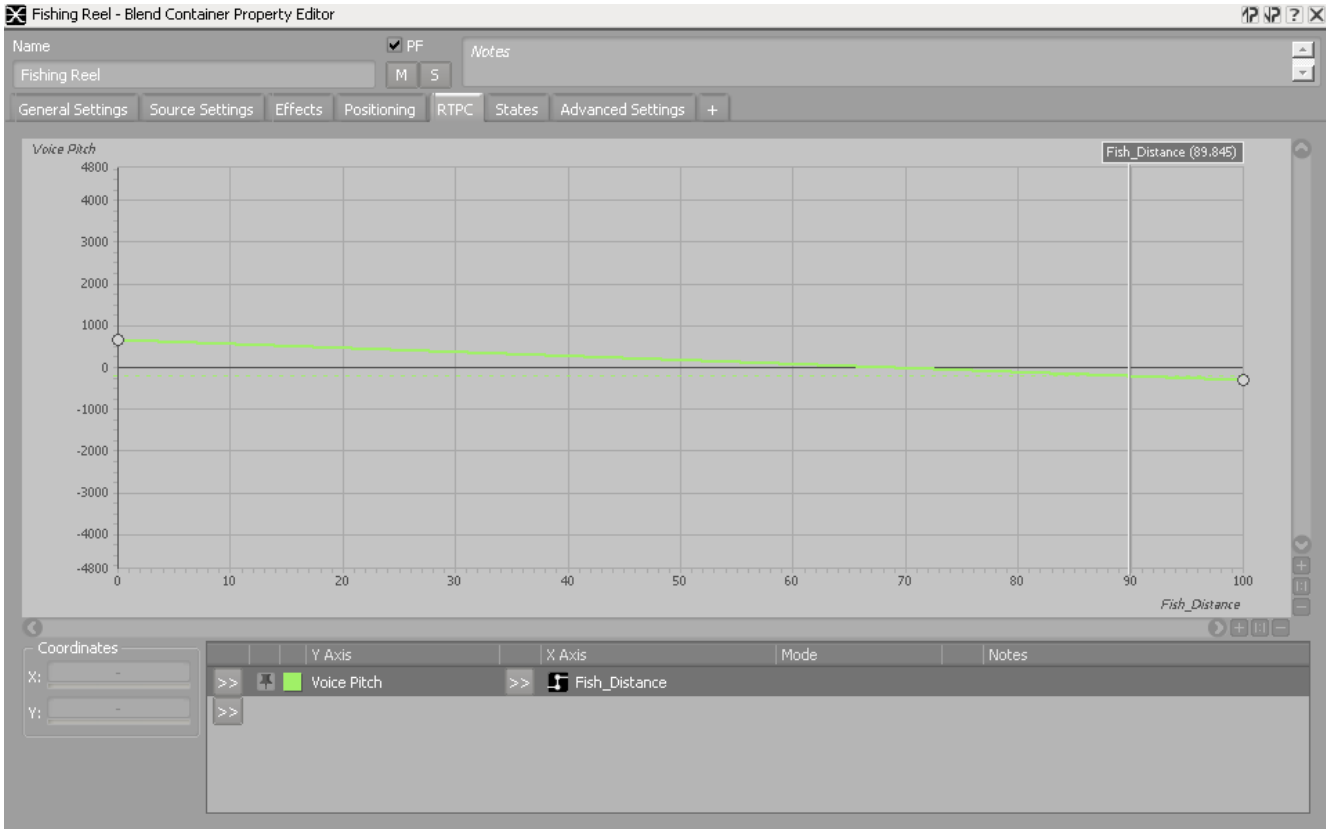
This is the script controlling introductory timings. The PostEvent("Testbeep") lines play in the order [Bar] -> [Beat] -> [Beat] -> [Beat]. From this point onwards, the script selects a random direction button, and then waits until the next beat to display a corresponding GUI icon. It then waits again for the next beat, where it colours the arrow green, and logs input from the controller. If the controller input matches the randomly selected input, then the player digs a bit deeper, and plays a 'digging' sound effect. Once the player has correctly several arrows in time, they will plant a tree in the ground.

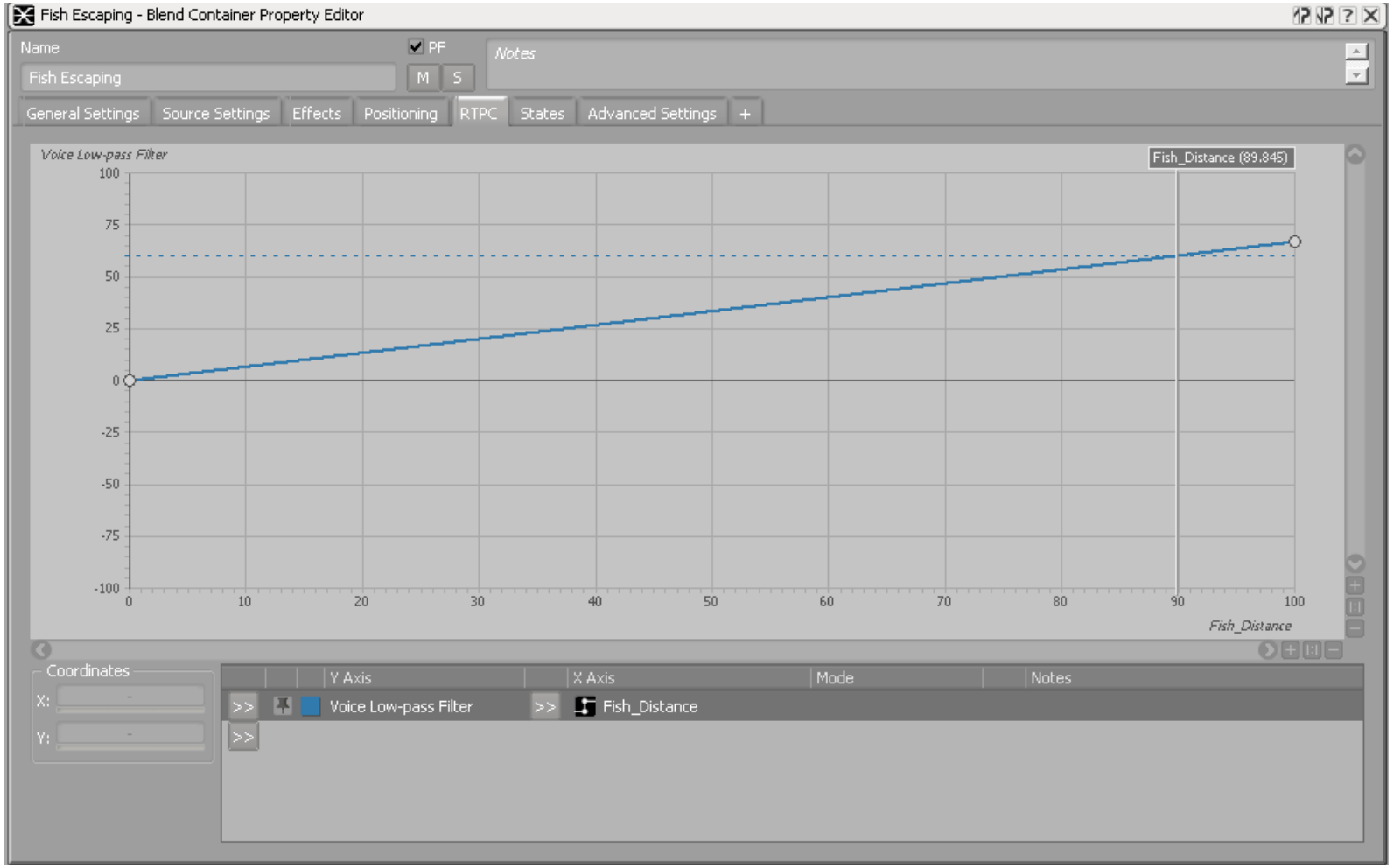
```
yield return new WaitForSeconds (0.1f);

    while (depthDug > 0) {

    while (!beatThisFrameRef)
    {yield return null;}

            yield return new WaitForSeconds (0.05f);
            randomButtonSelection = Random.Range (1, 5);
            switch (randomButtonSelection) {
            case 1:
                    selectedButtonString = "Down";
                    downImage.enabled = true;
                    break;
            case 2:
                    selectedButtonString = "Left";
                    leftImage.enabled = true;
                    break;
            case 3:
                    selectedButtonString = "Up";
                    upImage.enabled = true;
                    break;
            case 4:
                    selectedButtonString = "Right";
                    rightImage.enabled = true;
                    break;
            default:
                    selectedButtonString = "Empty";
                    break;
            }
```

```
yield return new WaitForSeconds(0.3f);
if(buttonLog.Count > 0){
    if(buttonLog[0] == selectedButtonString){
        if(buttonLog.Count >= 2){
        }
        else{
            depthDug = depthDug - 10;
            AkSoundEngine.PostEvent("Play_Digging", this.gameObject);
        }
    }
}
```

Fishing Interaction Sequence

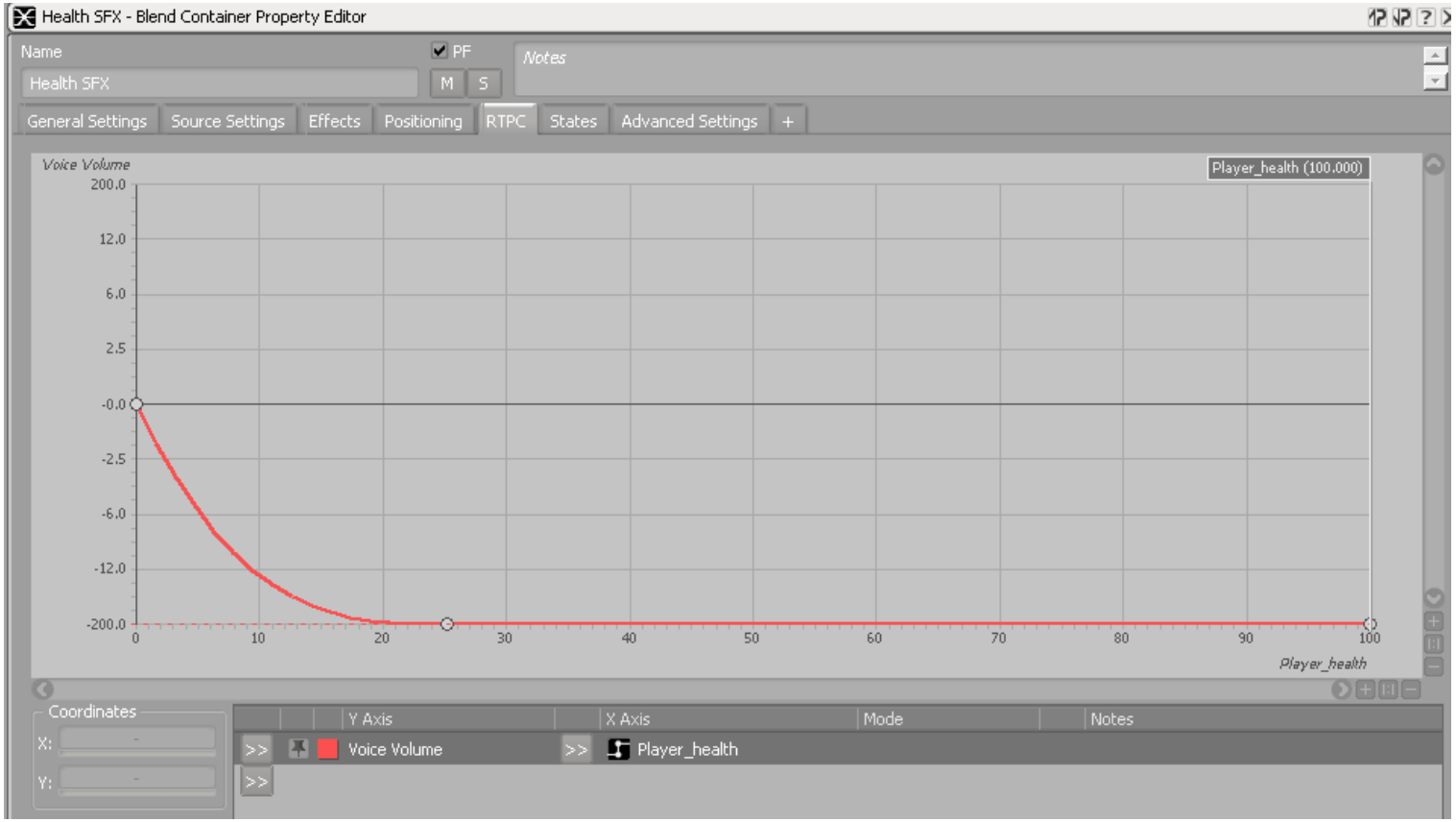
Similar to the digging sequence, the fishing sequence involves the player hitting a sequence of control inputs in time to the beat of music. However, in the fishing sequence, if the player mistimes, or mishits the input, the fish will move further away from being caught. This adds an additional element of difficulty, and requires the player to perfect their timing and hit many buttons in sequence. In terms of audio, this was represented in two ways. When the player successfully hits a direction, the sound of a fishing rod reeling in is played. The closer the fish gets to being caught, the higher in pitch this sound. Conversely, if the fail misses a beat, the sound of a fish splashing and swimming away plays, getting lower in pitch (sounding further away) as it gets further. This effect is controlled in real-time by using game parameters to change pitch, volume and filtering.

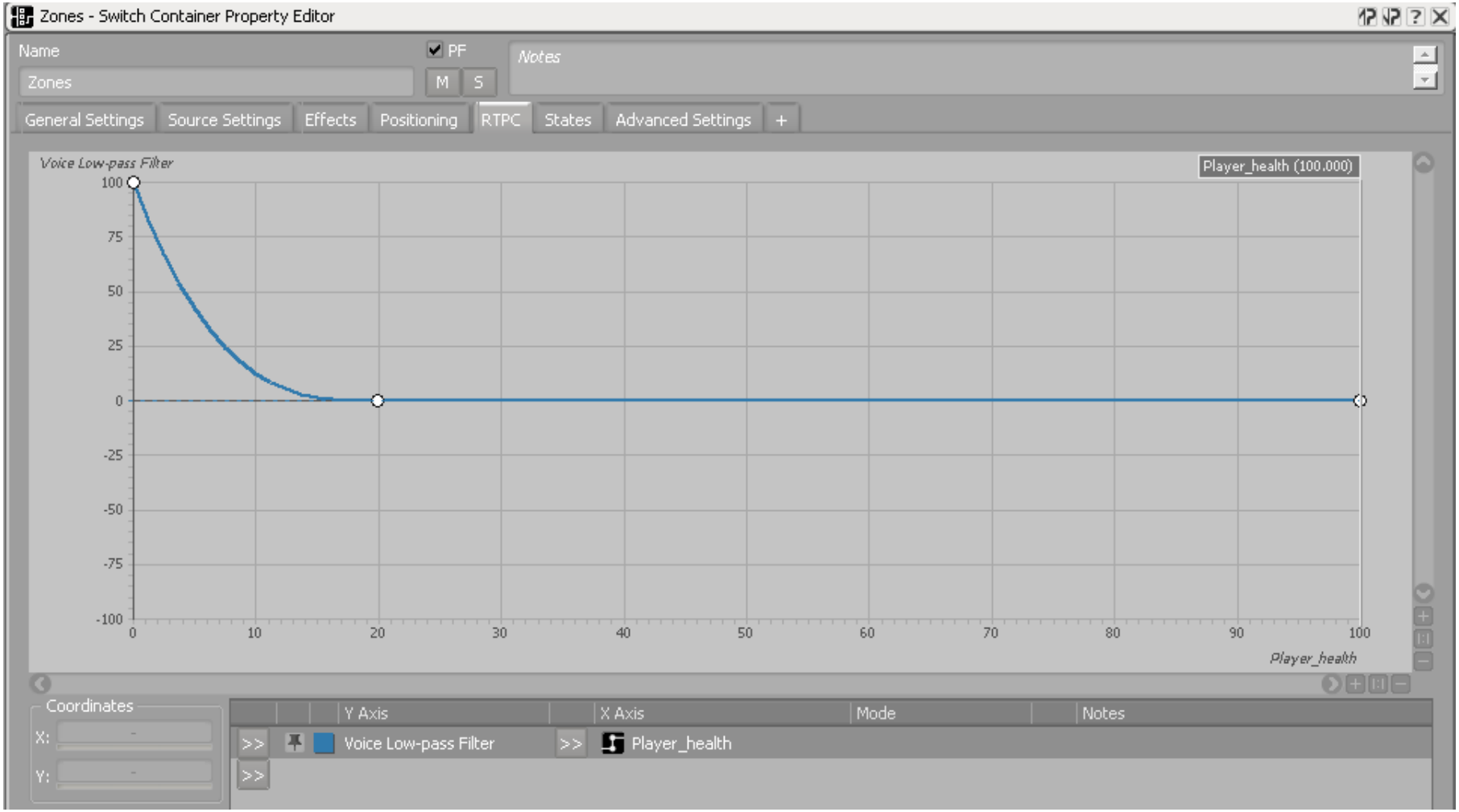



Fish Escaping - Blend Container Property Editor
Name
PF
Notes
Fish Escaping
M
S
General Settings
Source Settings
Effects
Positioning
RTPC
States
Advanced Settings
+
Voice Low-pass Filter
Fish_Distance (89.845)
100
75
50
25
0
-25
-50
-75
-100
0
10
20
30
40
50
60
70
80
90
100
Fish_Distance
Coordinates
X:
Y:
Y Axis
X Axis
Mode
Notes
Voice Low-pass Filter
Fish_Distance

Health Sound Effects

I wanted to represent the player's health level through sound and effects, to induce a sense of claustrophobia and paranoia when they are low on health. To do this, I created a game sync between Wwise and Unity which allows me to use the health level of the player as an audio effect. Using this game sync, I applied a low-pass filter to all in-game sounds, so that as the player becomes very low on health, so the soundscape they hear becomes less distinguishable. I also set up a synthesizer patch which plays a high-pitched sine tone, similar to the effect of tinnitus, which fades in as the player reaches these low health levels.



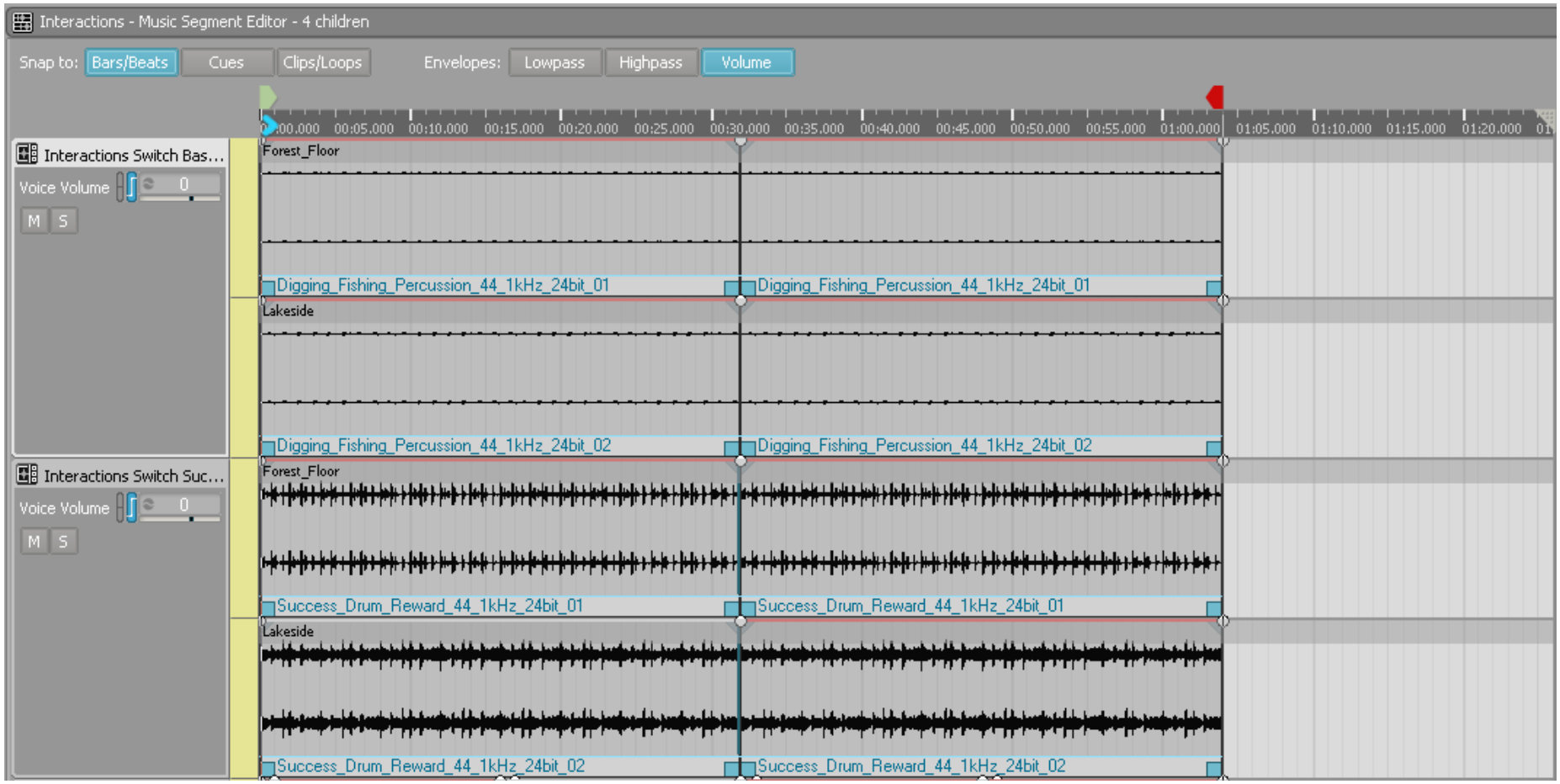

Zones - Switch Container Property Editor
Name
Zones
PF
M
S
Notes
General Settings
Source Settings
Effects
Positioning
RTPC
States
Advanced Settings
+
Voice Low-pass Filter
Player_health (100.000)
100
75
50
25
0
-25
-50
-75
-100
0
10
20
30
40
50
60
70
80
90
100
Player_health
Coordinates
X:
Y:
Y Axis
X Axis
Mode
Notes
Voice Low-pass Filter
Player_health

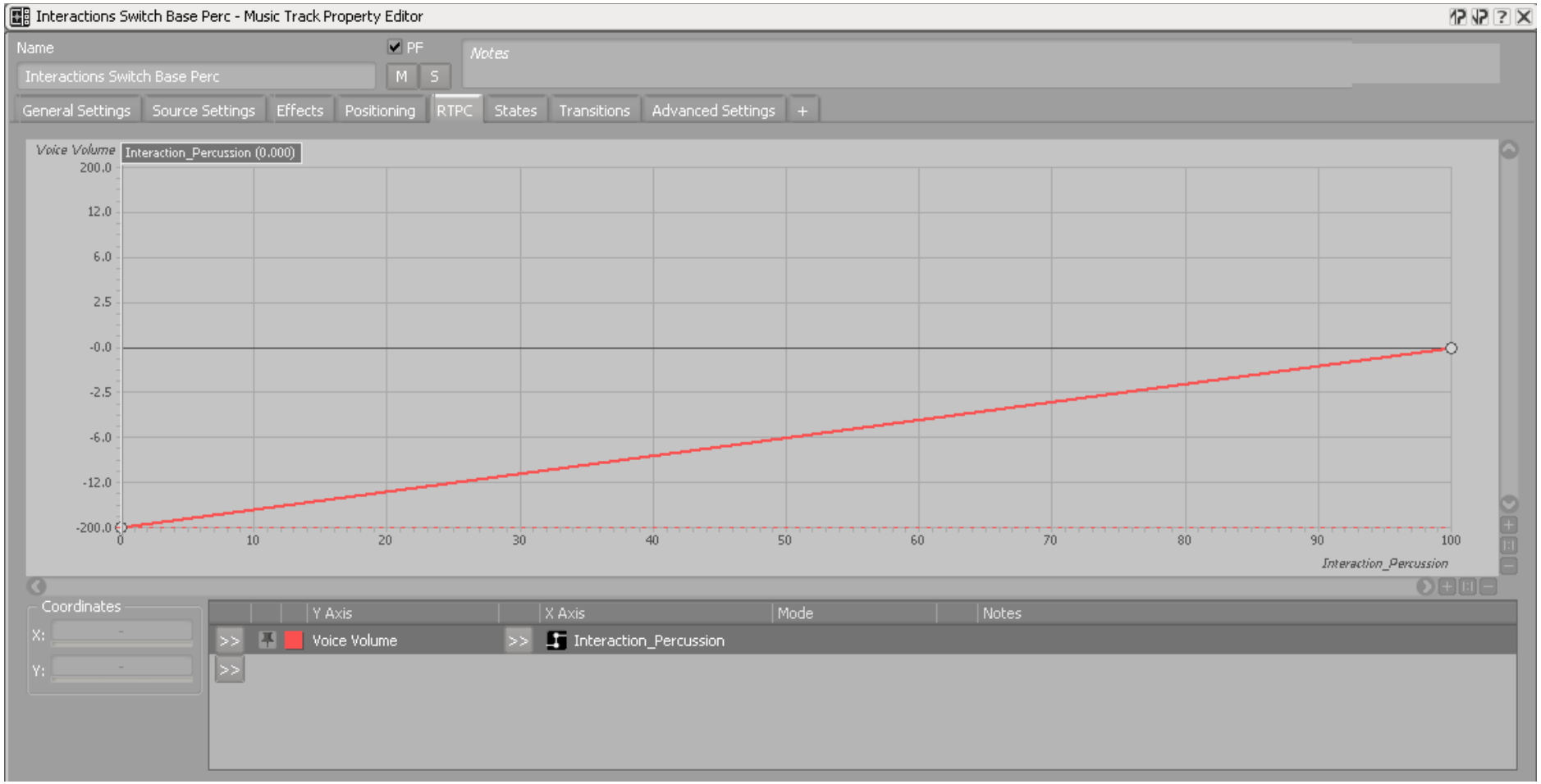
Interactive Music Implementation

Music in *Symbiosis* is controlled by the interactions of the player, and the subsequent effects these interactions have on the environment. Through the most basic interactions, the player influences the states of other elements of the ecosystem. The desire was that through consistent positive interaction with the environment, the music would grow, alternate and layer up to produce a constantly-evolving composition. A consistently negative, or passive approach to interaction, would have the opposite effect. As a result I decided to separate out the various elements of music and map each of these to various parameters of the game environment.

Percussive Rhythm and Simple Player Interaction

The most basic player interactions, such as digging and fishing, require the player to time button presses with the beat of music. As such, I decided to create simple drum tracks that would allow the player a foundation of timings that they could follow along to. These drum tracks mix in as the player decides to start planting trees, or catching fish. Additionally, if the player goes for a prolonged period without engaging with the environment, these tracks fade out. There are multiple drum tracks, each of which are mapped to different areas of the game world, so as the player moves about the space and interacts with it, so the music adapts and evolves.


Interactions - Music Segment Editor - 4 children
Snap to:
Bars/Beats
Cues
Clips/Loops
Envelopes:
Lowpass
Highpass
Volume
00:00.000
00:05.000
00:10.000
00:15.000
00:20.000
00:25.000
00:30.000
00:35.000
00:40.000
00:45.000
00:50.000
00:55.000
01:00.000
01:05.000
01:10.000
01:15.000
01:20.000
Interactions Switch Bas...
Voice Volume
0
M
S
Forest_Floor
Digging_Fishing_Percussion_44_1kHz_24bit_01
Digging_Fishing_Percussion_44_1kHz_24bit_01
Lakeside
Digging_Fishing_Percussion_44_1kHz_24bit_02
Digging_Fishing_Percussion_44_1kHz_24bit_02
Interactions Switch Suc...
Voice Volume
0
M
S
Forest_Floor
Success_Drum_Reward_44_1kHz_24bit_01
Success_Drum_Reward_44_1kHz_24bit_01
Lakeside
Success_Drum_Reward_44_1kHz_24bit_02
Success_Drum_Reward_44_1kHz_24bit_02

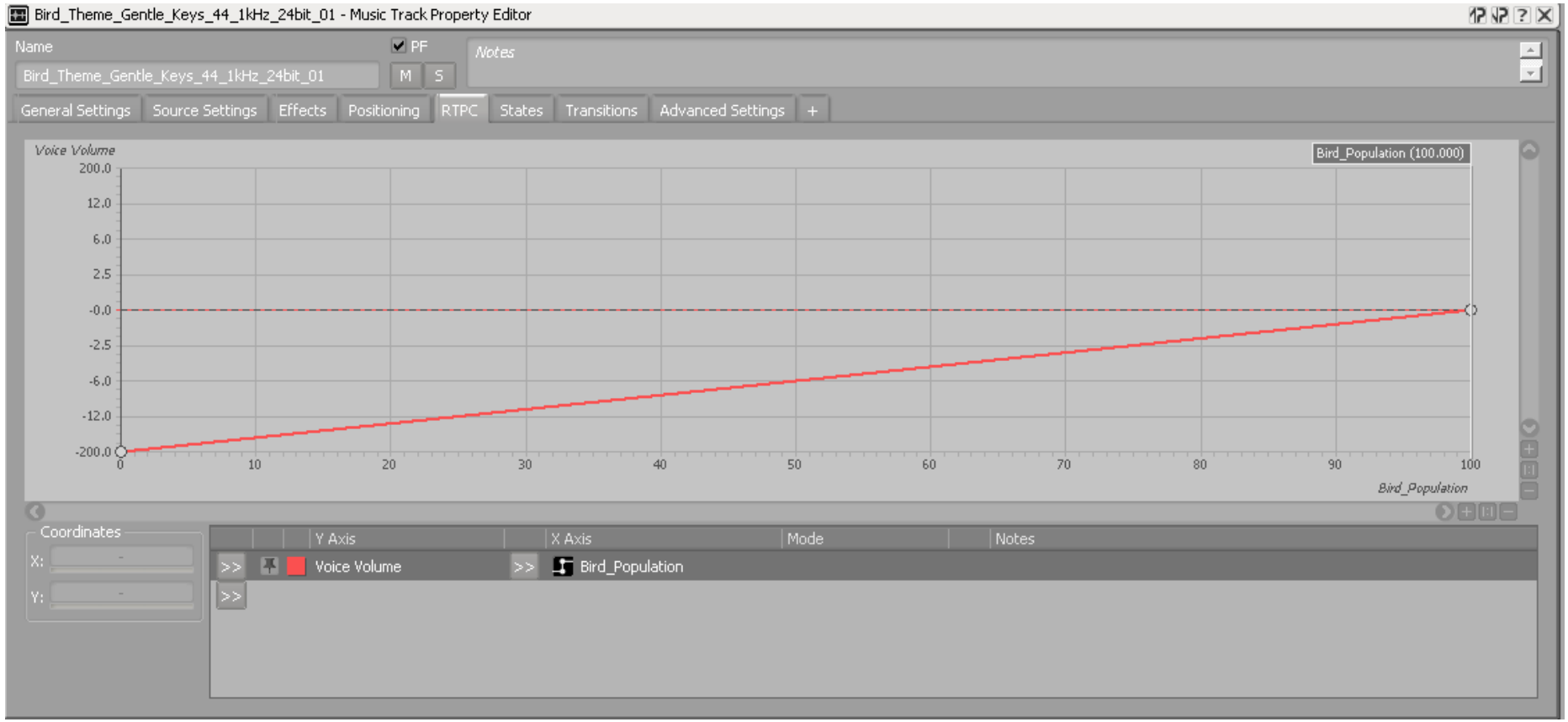

Interactions Switch Base Perc - Music Track Property Editor
Name
PF
Notes
Interactions Switch Base Perc
M
S
General Settings
Source Settings
Effects
Positioning
RTPC
States
Transitions
Advanced Settings
+
Voice Volume
Interaction_Percussion (0.000)
200.0
12.0
6.0
2.5
-0.0
-2.5
-6.0
-12.0
-200.0
0
10
20
30
40
50
60
70
80
90
100
Interaction_Percussion
Coordinates
X:
Y:
Y Axis
X Axis
Mode
Notes
Voice Volume
Interaction_Percussion

Sequences of Interaction and Bass Tracks

As the player begins interacting with the environment more frequently (e.g. planting plenty of trees), a bass track will mix in, reinforcing in the mind of the player that they are performing positive, forward-thinking actions. These tracks are also mapped to zones in the game world, adding further depth and variation to the music. Again, these tracks will dissipate if the player ceases to contribute to the environment.

Environmental Music and Species Mapping

In addition to the player-produced percussion and bass tracks, these are musical layers which are mapped to the states of the various species within the game ecosystem (birds, fish, wolves, trees, unique plant-life, etc). These musical layers were composed so they tonally and rhythmically sit well with the percussive and drum layers. However these layers are not directly controlled by the player's interactions. Instead, they are influenced by the population density, health and state of the species as a whole, which in turn can be influenced by player interaction. For example, the player may plant a lot of trees, which in turn will produce plenty of fruit. This abundance of fruit allows for growth in the bird population, which is represented through the presence of the musical layer assigned to the bird species. This level of detachment from the player's immediate actions allows for a great variety and depth in the musical soundscape.


Bird_Theme_Gentle_Keys_44_1kHz_24bit_01 - Music Track Property Editor
Name
PF
Notes
Bird_Theme_Gentle_Keys_44_1kHz_24bit_01
M
S
General Settings
Source Settings
Effects
Positioning
RTPC
States
Transitions
Advanced Settings
+
Voice Volume
Bird_Population (100.000)
200.0
12.0
6.0
2.5
-0.0
-2.5
-6.0
-12.0
-200.0
0
10
20
30
40
50
60
70
80
90
100
Bird_Population
Coordinates
X:
Y:
Y Axis
X Axis
Mode
Notes
Voice Volume
Bird_Population

Sound Design and Composition for *Symbiosis*

Compositional Style

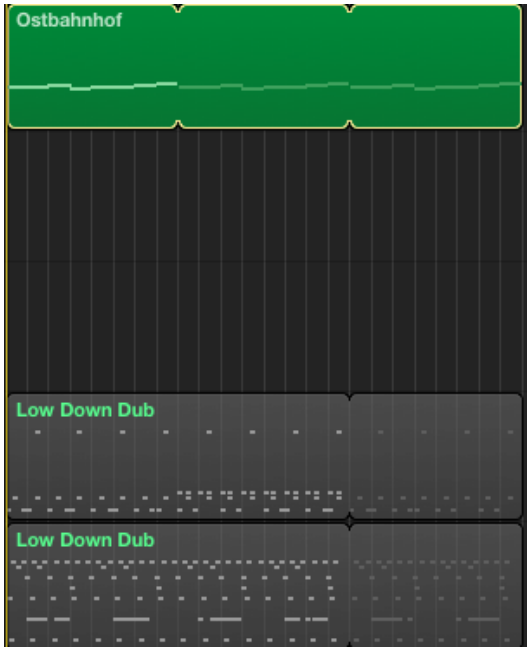
GAS

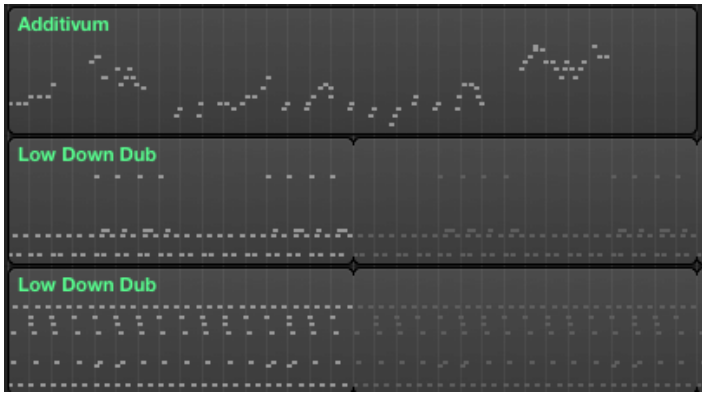
GAS is an alias for music producer Wolfgang Voigt, created during the late 1990s. During this period Voigt produced a number of albums which fused elements of ambient music and field recordings, with the dance floor aesthetics of 90s German club music. The music of GAS is densely layered, comprising of many segments of field recordings and pop music samples which were degraded, reversed, and stretched beyond recognition to create a broad palette of textures. Often underlying these textures and ambiences are four-to-the-floor drum beats and subtle drum machine rhythms which add an ebb and flow to the relatively amorphous ambient textures. Listening to the music of GAS could be described as hearing a party far in the distance of a dense forest, with only the throb of drums distinguishable from the conglomeration of woodland noises and creatures.

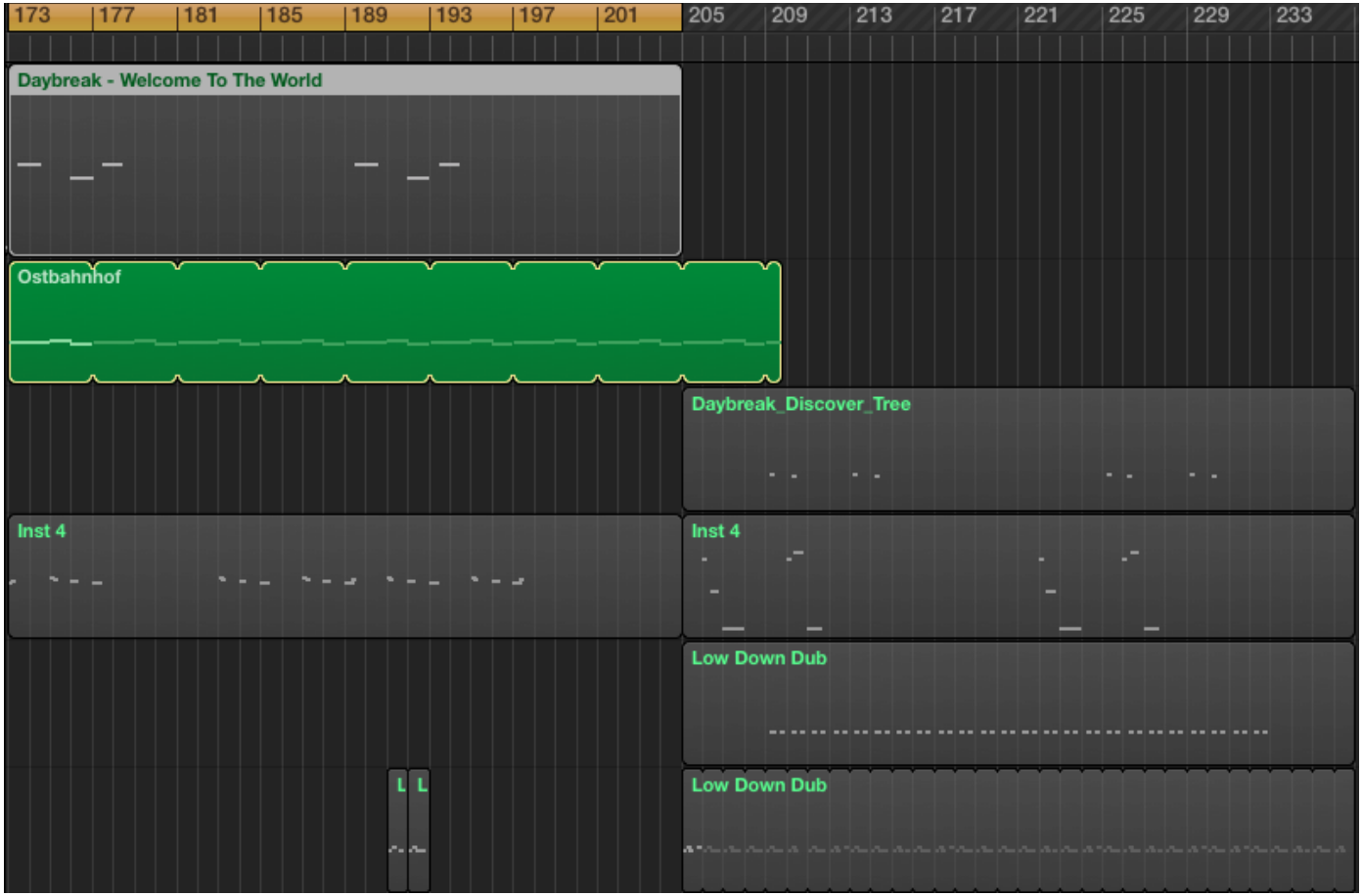
GAS inspired to some degree my compositional style for *Symbiosis*. The subtle layering and dense ambiences felt particularly suitable for the goals I wanted to achieve - to create an ambient/musical soundscape which can grow and evolve in subtle ways, yet also be stripped back to the noises in our immediate environment. Additionally, the aesthetics of the music suggest a fusing of natural environment, with electronics and synthetic noise, something which *Symbiosis*, by it's very nature of being a digital environment replicating nature, had already adopted.

Compositional Technique

Symbiosis features a score which can layer up dynamically as the player spends more time interacting with the environment. Since this score is dynamic - meaning that certain elements may enter or exit the composition at any given moment, I had to approach composing in a different fashion to linear media. I also had to be cognisant that some elements of the composition would be placed on game objects in the game world - i.e parts of the composition would play in non-diegetic space, whereas others would play in the diegetic world. As such, I primarily focused on composing loop-based music (which would also best serve the simple player interaction sequences). Individual tracks in these compositions were also designed so that they could transition between one another at any time. Originally, I had intended to have all of the organic objects in the game world be their own 'instrument', and that through the interactions of the player, these 'instruments' would play in different keys, with different tempos, and presence. Collectively, the entire ecosystem would then become a grand orchestra, which the player could explore and 'conduct'. Technically, this proved too challenging to achieve fully, however to some degree there are still objects which perform this instrumental role.





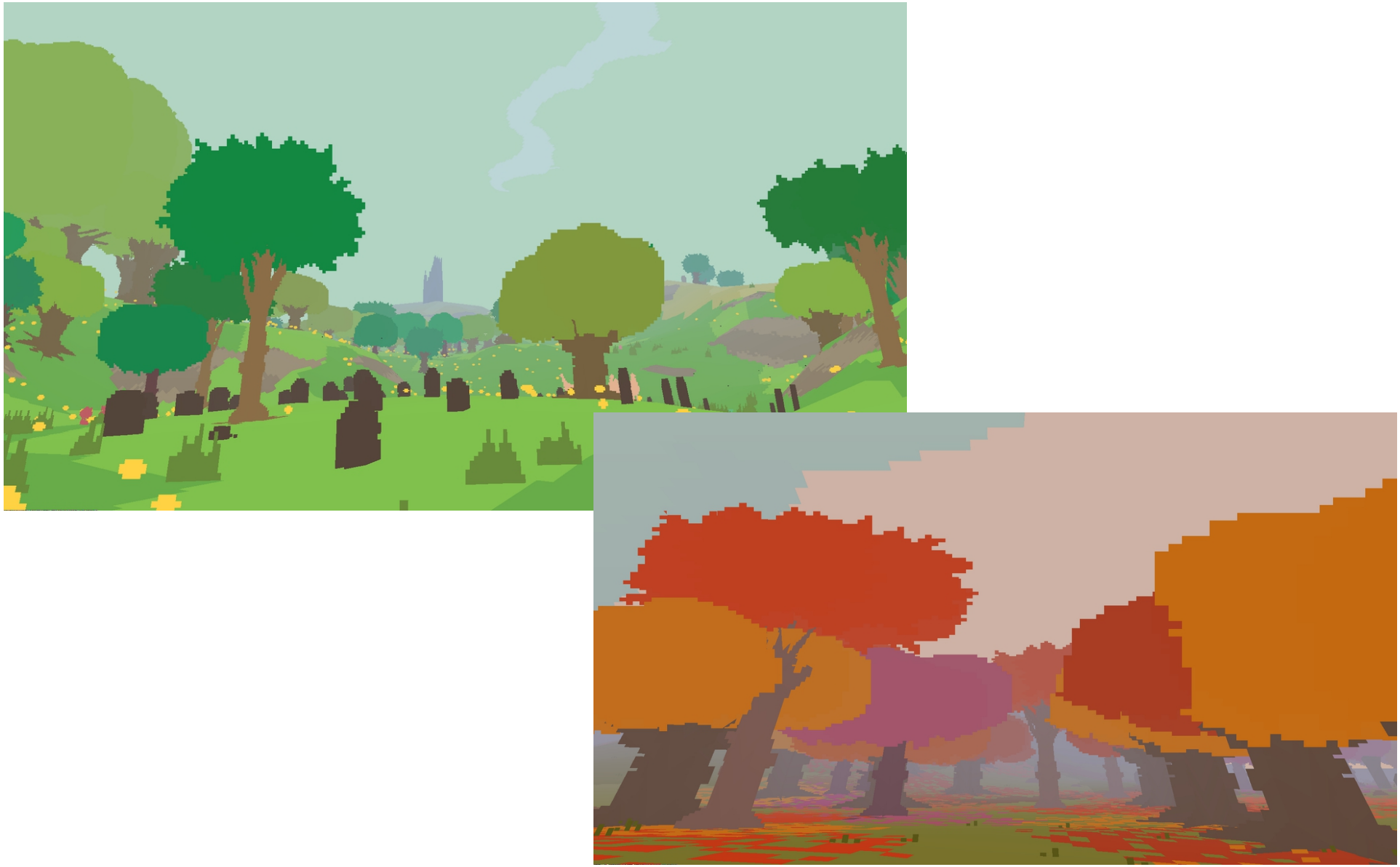

173
177
181
185
189
193
197
201
205
209
213
217
221
225
229
233
Daybreak - Welcome To The World
Ostbahnhof
Daybreak_Discover_Tree
Inst 4
Inst 4
Low Down Dub
Low Down Dub

Sound/Music as Gameplay, Research and Integration

A crucial theme in *Symbiosis* is how music and sound can be used as a mechanic of gameplay. In researching this topic, I looked at a variety of games, computer or otherwise, and how they attempted to use sound as a 'rule' in their gameplay systems. On the simplest of levels, I discovered games such as Musical Chairs, the children's party game. The rules of Musical Chairs dictate that the participants are to walk around a set of chairs whilst music is being played (e.g. from a stereo, instrument, etc). There is always to be one chair less than the number of participants. When the music stops playing, each participant must attempt to sit on one of the chairs. The person who is left without a chair to sit on is deemed 'out' and may not participate in further rounds. This game is an example of a very simple system where music is used to dictate gameplay. How it does this is by using the presence of music as a controller of how participants should be acting in the given moment. Whilst it is playing, players are able to jostle for the best positions on the chairs. When it ends (usually randomized), signals that the players should transition to sitting in the fastest way possible. In a sense, the game is testing our ability for focused listening, as well as reaction speeds, decision making, and movement.

Case Study: *Proteus (2013)*

Proteus is an open world exploration computer game developed by Ed Key and David Kanaga. The game world is comprised of procedurally generated islands which the player may explore. Each organic element on these islands have their own musical signatures, which come together to create beautiful, evolving music. In addition, the player has some degree of control over the time of day, seasons, and weather, each of which also influence the musical composition. However, other than this control of season/weather and the ability to explore the islands, there is little player interaction. The experience could be described as simple 'wanderlust', a desire to explore and encounter in a mostly passive manner. *Proteus* could be described as fitting the recent term 'art game', and has also been described as an 'anti-game', due to the perception that games should have rule systems and gameplay that test the player's abilities in a very active fashion. This game does not fully implement music as a gameplay mechanic, however the approach to the natural game environment as a musical entity which undergoes evolution, is one that I wanted to use in *Symbiosis*. The changing between moods and atmospheres in *Proteus* also became an inspiration for *Symbiosis*, and I wanted to incorporate a sense that the flourishing, healthy ecosystem would be producing beautiful, harmonious music.



Case Study: *140* (Dissertation Extract)

Many games make use of sound to convey complex information about game events to the player. In *140* this idea is applied to background music, where musical elements are synchronized to the movements of objects and enemies.[1] Music has been transmuted into the physical world of the game, and we must use our intuitive understanding of its rhythm to navigate. The game form comes in the shape of a two-dimensional *platform* game. Visual aesthetics can be considered very minimalist, with a vibrant colour palette and psychedelic feel.

Composer Jakob Schmidt created an electronic soundtrack which both complements the minimalist visuals, and works as a guide for player navigation. Heavy emphasis was placed upon rhythm, therefore music was composed in 4/4 time signature, 140bpm, and employs percussion, synthesized bass instruments, and half-time drum beats. Most significantly, the score was created with extensive game world integration in mind.

Objects in the environment that the player must navigate may appear, disappear, change state, or move, depending on what is occurring in the soundtrack. For example, a platform may appear ahead of the player as the hit of a snare drum plays in the music. On the next hit, the platform will disappear. As a platform game, *140* is very challenging, and requires players to execute their movements with very precise timing. The score is therefore providing an intuitive system by which a player is able to very accurately gauge how and when the environment will change its status.

1 K. Aallouche, H. Albeiriss, R. Zarghoune, J. Arrasvuori, A. Eronen, J. Holm, 'Implementation and evaluation of a background music reactive game', In Proceedings of the 4th Australasian conference on Interactive entertainment (IE '07), M. Gibbs, Y. Pisan (eds.), RMIT University, Melbourne, Australia, Article 1.

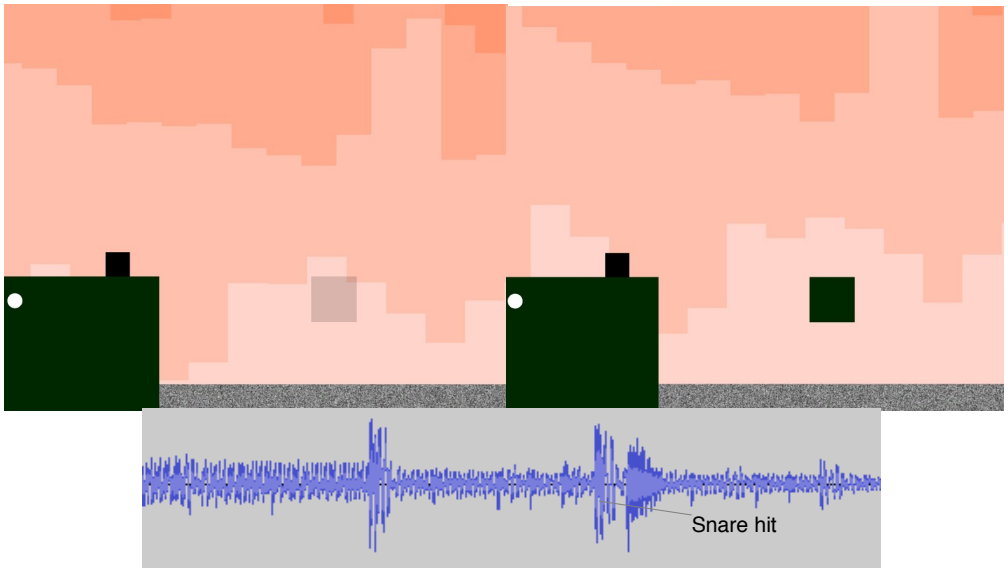


Fig. 3: Appearing and disappearing of platform on subsequent snare drum hits.

Similar to forms of electronic dance music, there is heavy use of repetition and *looping*. This allows the player to become accustomed to the musical elements and how they relate to game objects. Progressing through the levels is also linked to musical progression. Instruments come in as the player encounters their respective objects, and fade out as they move past them. As an experience this could metaphorically be likened to playing within a piece of music.

In the first levels the synchronization between instruments and objects is readily apparent and easy to follow. However, the second 'boss' level the player encounters involves avoiding an enemy whose attacks are timed to the playing of a bass note. In order to successfully dodge the attacks, the player must understand that it coincides with the third beat of every second bar. Through kinesonic synchresis, the player is able to forge a connection between the bass note, and the repeating attacks of the enemy. The player must dodge 12 of these attacks in order to beat the boss, so a mastery of timing is imperative.

The final boss battle demands a much greater understanding of the music than the preceding levels. Whereas previous levels required the player to keep time with a single musical element, this battle involves multiple. It also introduces a set of random conditions which prevent the player from simply memorizing a set of physical actions. Instead they must keep track of a snare drum pattern, which determines the position that their character fires from. They must also pay attention to the timbre of a synthesizer chord, which represents the direction that enemies will come from. This level also overloads the visual system by rotating the game world during each chord. Visually orientating where the character needs to be is practically impossible. Instead, the player must pay close attention to the snare pattern and synthesizer chords in order to master the level.

140 makes significant use of multimodal sound. The soundtrack is capable of conveying detailed information about game action, which the player must be attentive to in order to succeed. Visual representations of the game action are insufficient to make the game enjoyable and immersive. As Sanders and McCormick describe, auditory communication is preferable when the visual system is overburdened, or subject to continuous change.[2] [3] The complex movements and interactions of game objects are much better conveyed through musical timings. Gameplay as an experience involves extended periods of *signal* listening. Frequently the player will pause for a moment, until they have an understanding of musical timings and how they relate to objects.

2 E.J. McCormick, M.S. Sanders, *Human Factors in Engineering and Design: Auditory, Tactual, and Olfactory Displays,* McGraw-Hill, Singapore, 1993, p. 133.
3 K. Jørgensen, op. Cit., p. 62.

Case Study: *Fract OSC* (2014)

Fract OSC is an open world, music based, puzzle solving game developed by Richard Flanagan. The visually abstract world contains many objects or 'organisms' which respond to player interactions with sound and music. Throughout the world are scattered short puzzles which involve the player rearranging, activating or modulating these musical objects in order to achieve certain sounds. This world could be thought of as a large-scale, explorable synthesizer, where the puzzles require the player to match a predefined composition by altering the synthesizer parameters. *Fract OSC* is therefore an excellent example of using sound as a core component in the gameplay experience. The sound forms part of the rule system in which the player must work, and challenge is found in how well the player can rearrange the current sound, with that of an 'ideal' sound. This idea was partially incorporated into *Symbiosis* - through the mechanics of interaction that involve fishing and planting trees.

4

4 Image ref: *Fract OSC* (2014), Richard E. Flannagan

Written Notes of Texts and Articles

Bridgett, Rob. *From The Shadows Of Film Sound*. R. Bridgett self-published, Vancouver, 2010.

P.10 - Around 2004, high-profile film voice actors, composers, and sound designers started working with game development teams. This was as part of a mutual recognition of each other's crafts as artistically valid.

P.11 - Games began to adopt the same audio technology as film (HD audio, surround sound, etc) made possible by DVD/Blu-Ray.

Major work - Danny Elfman: *Fable*

P.14 - Bill Brown (*Lord Of The Rings*, *Hulk*, *CSI*): "Game music is becoming much more important in working to support the narrative of games. Developers are very interested in the dimension that music brings to a game."

P.15 - Big-name composers are given much greater public recognition.

P.16 - Film music and game music are aesthetically close. However there are vital differences:

- Film and TV, the score is 'locked' to the visuals. Repetition can be advantageous as the score will only be heard once. In games, repetition can become tiresome. Both can be considered to be enhancing the emotional experience.

- Films are 'static', they follow linear timelines. Games are 'dynamic', they alternate between many different states and often repeat themselves.

P.18 - Even the formats for delivery can be very similar.

P.19 - How a game will implement audio is crucial to how music is composed for it. It is not possible to write/implement music for every possible game scenario.

P.20 - Composers, sound designers, etc, work closely with game production teams in a very similar fashion to film crews.

P.21 - The overarching 'musical language', i.e. the aesthetics of sound in relation to all other aspects of the game, and how communication takes place, is imperative to both game and film.

P.24 - Music in film is well defined by the films stylistic aesthetics, as well as the visual edit. It is important for composers to be aware of the underlying systems in a game in order for them to create a high quality set of interactive audio. In film, scene edits help dictate the length of a piece of music, as well as pace/tempo, derived from the degree of acton in the scene.

P.25 - Key components of interactive music's micro-structure:

- Narrative Cues: Linear, predictable, i.e cutscenes. Music functions in the same way as in film.

- Continual Cues: Music that keeps playing until the player interrupts. Such as menu music, journey music, etc.

- Evolving Cues: Music that responds to changes in the game system. For example, if a player is in stealth, and they are caught out by an enemy, the music may transition from 'Stealthy Music' to 'Danger Music'.

P.64 - From Film to Games

P.71 - Off-screen sound, particularly in games that employ periods of visual darkness, become functional in assisting the player in navigation, object/creature identification, and threat analysis.

- More can be done to make use of off-screen sounds beyond 'dark' games.

P.73 - GUI readouts could be sonified in more interesting ways. Subtle changes in musical tempo or liveliness could be linked to a player's health, for example.

P.80 - Diegetic sounds and music in games are usually easy to recognize - the sounds are positioned spatially in the 3D world somewhere. Reverberation is often present.

- Non-diegetic sound is often unprocessed and locks 3D positioning or reverb.

P.81 - Being able to interact with the sources of diegetic sound reinforces immersion. E.g. scaring off a noisy crow or turning off a radio.

- Hearing a police scanner responding to player action (*Grand Theft Auto*) further reinforces diegetic immersion. [Note: repetition of the same reports can hinder it]

P.85 - As Pierre Schaeffer and Michel Chion noted in regards to 'reduced listening', audiences are often overwhelmed by a desire to supply mental visual imagery as to the source of a sound, when few or no on-screen visual clues are given.

Collins, Karen. 'An Introduction To The Participatory And Non-Linear Aspects Of Video Games Audio', *Gamesound*. Last updated 2014. Accessed October 2, 2014.

P.1 - The music of video games differs from that of traditional linear music in that it can start, end, or transition at almost any time. A comparison can be drawn where linear music is a straight, direct-route train between stations, whilst the non-linear music of games is more like a city underground rail network, where there are many beginnings and endings, with a near infinite number of possible routes to take.

- As a result, compositional style for game music differs significantly to that of other linear media, such as film or TV. Composers are often required to create much shorter, less synchronized, highly versatile musical 'elements', that are capable of transitioning into one another, without sounding disjointed.

P.2 - Interactive audio can be defined as sound events that occur in respect to gameplay and the direct input of the player, For example, the shooting of a gun results in a gunshot sound.

- Adaptive audio refers to sound events which "react appropriately to - and even anticipates gameplay", rather than responding directly to the user.

P.3 - Dynamic audio, such as the above, complicates the diegetic/non-diegetic relationships between sound and moving image, as was known in film and television. Sound which is initially diegetic, could easily become non-diegetic through events in gameplay. As a result, these broad categories of diegesis and dynamism must be thought of in a fluid, or temporal sense, constantly shifting.

- The non-linear and interactive nature of game sound can be thought of to reflect a post-structural theory of narrative, a "return of the reader" and "death of the author" form.

P.5 - Some methods of transitioning between musical elements include crossfading, which can be jarring if there is a major emotional change in narrative, and 'stingers' which mask a transition with an abrupt motif, which is especially useful for transitioning into combat. Cue-to-cue transitions, where musical playback is delayed until the next musically appropriate moment, and layering, whereby musical elements are intended to blend seamlessly into a whole soundscape, are more recent techniques.

P.6 - Stockburger (2005) identifies five different categories of sounds:

- Dialogue
- Ambience
- Score
- Diegetic Sound Objects
- Interface Sound Objects

- Anahid Kassabian raised the issue of the "evaporating segregation of sound, noise and music", and how the music of games has influenced film (e.g. *The Matrix*), whee the difference between musical score and sound effect is greatly reduced.

P.7 - Gameplay time is completely unpredictable. How, therefore, does music that must lack a beginning, middle, or end, function?

- Listener fatigue is also a vital consideration when scoring, as players may spend very long amounts of time listening to the soundscape of a certain zone or level.

P.8 - A crucial function of sound in games is the preparatory role, where sound can foreshadow an upcoming game event, allowing the player time to prepare. Anticipating action is crucial to many games. Most notably is acousmatic sound - that with no clear visual origin, which inspires us to look and go there (Chion, 1994). Sound symbols or leitmotifs can help direct our attention towards important game events. E.g. lesser enemies may have the same music cues, greater enemies may have more unique or attention grabbing cues, assisting the player in scaling the learning curve.

- Cohen says a lullaby playing over an image of a room with a cradle and a fishbowl will direct our attention towards the cradle.

- Boredom switches (a simple game mechanic that fades out music in a game level once it has been listened to for too long), can help inform the player that they have completed a level, or that the game is waiting for the player to progress.

P.9 - Dialogue can be extremely important in guiding a player through narrative, and help them make effective decisions. Not only in the implicit sense (a character telling the player what they need to do), but also more hidden suggestions. For example a Non-Player Character (NPC) discussing world events with other NPCs can hint that an enemy may be at a certain location.

P.10 - Unlike film and television, where score is "designed to be absorbed on a subconscious level as 'unheard melodies'" (Gorbman), audio in games often requires active listening, as the suggestions the music makes to the player is critical to guiding their decision-making process.

Jørgensen, Kristine. *A Comprehensive Study Of Sound In Computer Games*. Edwin Mellen Press, Lewiston, 2009.

P.2 - "I realized I was aware of incoming enemies before they were onscreen" in reference to leitmotifs (Gorbman, 1987)

P.4 - Sound has two over-arching purposes in games - to act as a usability system to convey information to the player, and as a "convincing" agent to solidify the believability of the game world.

P.8 - Games of different genres are likely to employ sound in very different ways.

P.9 - A game such as *Hitman* features an in-game player avatar, and therefore communication can occur between game world and player, through diegesis. Sound also takes on a 'realism' role to convincingly portray a modern world. *Warcraft III* has no in-game avatar and therefore sound takes on a less naturalistic role, with the game world communicating directly to the player.

P.14 - The first 'authentic' computer game did not feature sound in it's playable release version, although earlier test versions did, suggesting that sound was considered important to these early computer games.

P.17 - *Pong* (1972) featured functional sound, informing players when the ball had bounced and therefore acted as a cue for player attention.

- *Space Invaders (*1978) had more remarkable audio, with music which increased in tempo as invaders get closer to the player. This was the first game to use non-diegetic sound to affect the player's engagement with the game.

P.20 - *Super Mario Bros* (1985) used sound for "urgency and response purposes". E.g. the 'Urgent' or 'Hurry Up' music, or when the sound of a cannon firing is heard from off-screen, shortly before a bullet appears.

- *The Legend Of Zelda* and *Final Fantasy* used unique music in dungeons and areas with particularly powerful monsters.

- In the late '80s and early '90s, specialized sound cards which afforded greater freedom to developers became commercialized. *Wolfenstein 3D* (1992) employed stereo panning which was used to indicate the movements of sound sources relative to the player.

- *Duke Nukem 3D* used real-time audio effects to simulate virtual environments.

- "These titles pointed out new uses of game audio beside the usability function, connected to orientation and sense of presence".

P.23 - *Thief* uses voice and footsteps from other characters to signal enemy presence and alertness towards the player.

- *Tomb Raider Legend* uses adaptive music to alert the player to dangerous situations, and their proximity.

- The origin of game audio is two-fold: First, it is the pure functional sound. Second, it it is the role of creating a sense of space for action to occur.

P.24 - Computer games consist of two layers: the surface layers of audiovisual stimulus, and the underlying systems. In order words - the game as rule system, and the game as imaginary world (Jesper Juul, 1999, 2005).

- Lars Konzack delineates two areas to consider in game analysis: 'virtual space' and 'playground'. Virtual space is the space of action in the game, with characters etc. The playground is the means of interface with this space (Konzack, 2002).

P.35 - Stockburger sets out classifications for game sound:

- Score Sound Objects
- Zone Sound Objects
- Speech Sound Objects
- Interface Sound Objects
- Various Sound Effect Objects (related to the avatar, other game characters, etc)

P.38 - Juul defines game elements in terms of emergence and progression. Progression involves the player performing a specific set of actions in order to complete a given set of objectives, in order to advance. Emergence means a simple set of rules, through which an extreme variety of outcomes can be produced (2005).

P. 44 - Sound can be used in the absence of haptic perception, i.e. taking a gunshot to the leg.

- Volume of footsteps when walking compared to running inform the player as to their level of stealth and audibility, which is particularly relevant to games that involve enemies.

P.52 - Actions in games can comprise of three different fundamental modes of interaction:

- Activities can be any kind of participation that is active. E.g. hypothesizing, turning a page in a novel, operating a controller. These do not directly progress the game (but in combination, they can).
- Simple actions are those of participation, which are meaningful and are realized as physical actions in-game, but do not contribute to the progression. E.g. exploration, modification.
- Progressive actions, which can be the same as simple actions, but have a direct effect on the progression of the game.

P.60 - In many aspects, it is not 'realism' that sound designers are striving for, but 'credibility', the believability of the game world. Perceptual fidelity would be the goal.

- Functional fidelity refers to how well a sound performs it's role.

P.62 - Sanders (1986) lists areas in which audio can be preferable to visuals in conveying information. The most important of which is when our visual system is overburdened or hindered. Also, when the message is short, or subject to continuous change. Other situations include alerts, warnings, etc.

- Sound can therefore be employed proactively (autonomous delivery systems), or reactively (in response to user input).

P.69 - Langkjaer points out that sounds often reveal their origin in films, but that this is not of great importance to the audience. "Instead it is often more important to understand the situation, or whole context that sound and image appear in". I.e. the sound of a gong in *Warcraft III*. It isn't important to know where the sound originates from, but that a change in status has occurred.

P.70 - Gibson notes on the 'affordances' of objects, meaning that any object we meet will present us with ways we may interact with it. In game contexts, this means a monster may present us with information as to whether we can run away, or fight it.

P.92 - In Auditory Display studies, there is a focus towards two important functions: the use of sound for urgency purposes and for responsive purposes. Urgency is self-explanatory, but can be separated into three distinct categories - highest, secondary and tertiary priority. The higher the priority, the more distinct and informative the sound should be. Responsive refers to the human-computer interaction paradigm, where sound is often employed to provide confirmations and responses to user actions within computer systems. These can be subcatagorized into positive and negative feedback.

P.97 - Transdiegetic can be understood "as the conceptual space that comes into being when communication in a game questions the boundaries of the virtual world".

P.98 - The most fundamental difference between film and game audio is the effect that sounds may have on the internal diegesis. Non-diegetic can indeed effect what happens in the game world. This is transdiegeticism.

Collins, Karen. *Playing with Sound: A Theory of Interacting with Sound and Music in Video Games.* MIT Press, Massachusetts, 2013.

P.3 - Walter Murch describes how many sound effects in live-action movies are 'sound-centaurs', half-music, half-language. This is especially true in games.

P.5 - A fundamental distinction: we listen **to**, we interact **with**. Does interacting change our subjective experience? If so, how?
- David Huron lists expanded listening modes (beyond Chion). Of particular relevance to games are:
    - Signal listening which involves waiting in anticipation for a certain cue.
    - Retentive listening which involves memorization of information.
    - Sing-along listening which involves mentally or physically following along with a sound.
- These are all interactive modes of listening.
- Other ways in which we can interact with sound include evoking, selecting, shaping, or creating.

P.7 - There is significant contention over what it means to interact with a media form.
- Morse states that "interactivity involves decision-making or the active participation of the user"
- Many theorists "place agency and the ability of the media to respond physically to the audience's action as central elements of interactivity".

P.9 - Cognitive/psychological interactions do not directly involve physical control or feedback, however such interactions occur alongside all other types of interactions ingame.
- Psychological interaction exists only with the interaction of the body.
- An embodied interaction with games is key to understanding their psychological effects, too.
- Some definitions refer only to physical interaction - mostly inadequate.

P.12 - Interactive texts require active participation to be complete.

P.16 - Embodied cognition "holds that our understanding of the world is shaped by our ability to interact with it".
- Embodied cognition is "the creation, manipulation and sharing of meaning through engaged interaction with artifacts". (Dourish 2001)

P.17 - In this theory, our cognitive processes use reactivations of sensory and motor states from our past experience.

P.19 - 'Interactivity' was coined after a century of largely passive media experiences.

P.20 - What is new about interactivity is the technology that mediates the experience and it's ability to fundamentally alter the ways in which we interact with media.
- Technology allows for disembodied sound.

P.21 - This allows "corporeally sensitive" listening modes. We don't just see sounds occuring; we feel them.
- Synchresis (Chion) is the fusion of sound and image that may create new meanings that alter or add to the original.

P.22 - Studies over embodied film theory are almost completely ocular-centric. The same can be said for video game studies.

P.25 - In simple terms, the players stakes are higher for their involvement, interpretation, and therefore attention. Players therefore listen more actively and employ different modes of listening to guide their movements and actions in games.
- Film may act on the body, players act with games.
- Sound in interactive media is multimodal.
- Vision, audition and haptics all combine. In fact, interactive sound almost certainly requires more than one modality.
- We can never not listen causally - it is the dominant mode of hearing.

- Leman contends that music involves all of the senses.
- Music "moves the body, evokes emotional responses, and generates associations with spaces and textures".
- Sounds are by their nature, semiotically loaded with references to other modalities. E.g. percussive hits connote movement or gesture.
- Sounds we dislike most could be related to our haptic associations (e.g. nails on chalkboard) (Trevor, Cox, 2008)

P.24 - Radio drama unfolds in the mind through mental imagery.
- Sounds of footsteps approaching serve as an index for the event (someone is approaching) but also carry wider associations (e.g. intent, gender, size, etc).
- *AudioDoom*. We can create a physical game space in our mind through spatial audio.

P.25 - *Papa Sangre* requires bodily engagement and active listening.
- The sounds without images are not disembodied because they carry corporeal, haptic, and visual associations (Chion).

P.26 - Studies on synchresis usually describe how audio and visual can be congruent, incongruent, or neutral.
- Theories on multisensory experiences indicate that there is a binding of information, and emergent meaning, that could not have been obtained from each modality on it's own.
- Walter Murch: "we do not hear or see a film, we hear/see it".

P.29 - Multimodalities can reduce our cognitive load, particularly where information is dense or complex.
- They can also increase it, where false information is presented (e.g. Stroop task)

P.32 - We can conceive of a 'kinesonic synchresis' where action and sound may also interact to produce emergent meanings.
- Interactive sound in games is kinesonically synchretic; sounds are fused to action, image, and even haptics and gesture.
- Sound is event-driven, controlled by an action or occurrence that is initiated in the game or the player.

P.33 - Sounds are commonly used as feedback to acknowledge an event. The relationship between input and output is likely to be synchronous.

P.35 - Sound may also be congruent, incongruent or neutral in relation to player action.

P.39 - The same neurons fire when either observing, or performing physical actions.
- Our emotional and neurophysical states can be directly affected by what we see.

P.40 - India Morrison and Tom Ziemke (2005) argue that this mirror response applies to virtual characters, too.

P.41 - Studies have shown that the brain will respond to game controllers as an extended part of the body.

P.42 - We are able to extend our sense of self into a virtual environment through the senses.

P.43 - Sound in games can extend our body into an intermediary space between us and the virtual world.
- "Sounds that are self-produced provide us with important feedback about the world we live in, and help to delineate our sense of self"

P.44 - "From early in our lives we use sonic feedback to determine whether something is a part of ourselves, or something external" (Rochat, 1995)
- Players cognitively perceive self-produced sound as "their" sound.

P.46 - Deacousmatizing is an important component in games like *Dead Space*.

P.47 - Sound is responsible for creating the 3D presence.

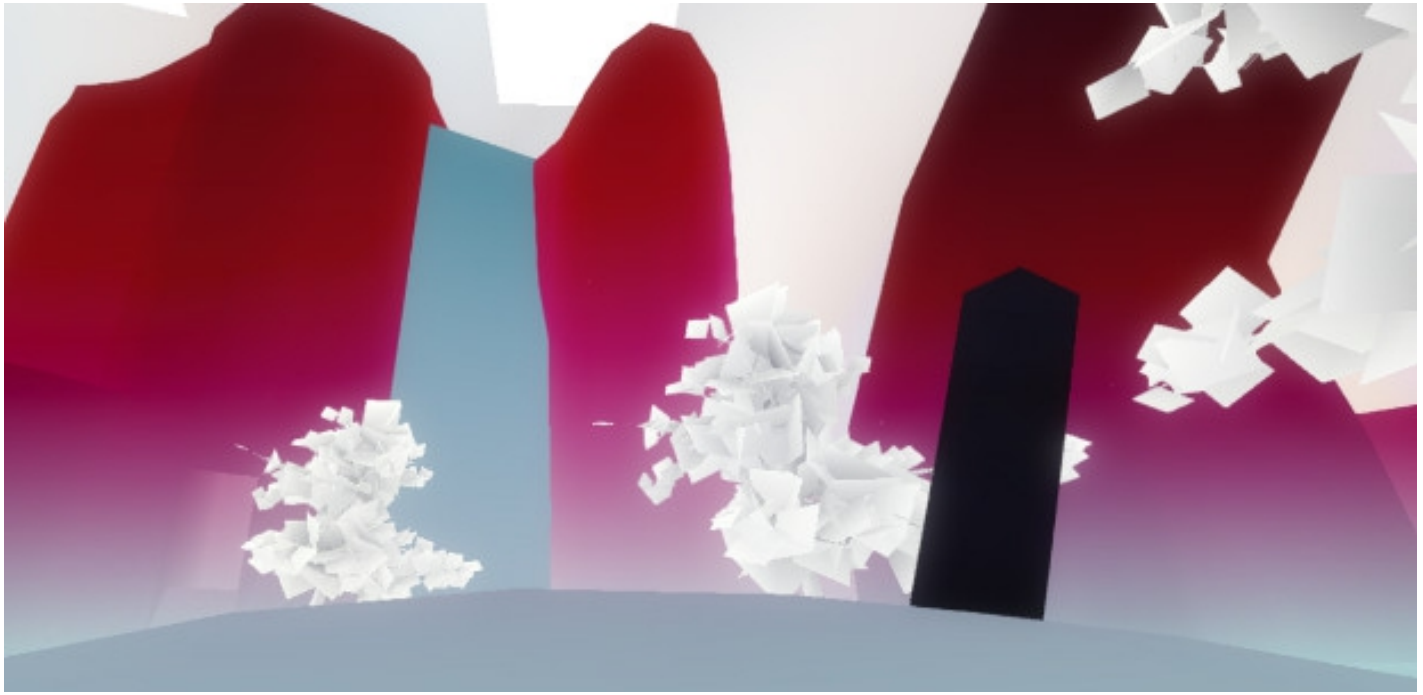
P.48 - Perspective in games is fundamentally different from film as the player has full control over character and camera.

P.53 - Envelopment in sound has been shown to be very important in creating a sense of 'presence', that is, to reduce the perceptual distance between audience and virtual space (Rumsey, 2002)(Berg, 2009).

P.58 - Rick Altman comments on how sound "constitutes the perfect interpellation, for it inserts us into the narrative at the very intersection of the two spaces (1992).

Case Study: *The Bends*

The Bends is an exploration and music game in which the player explores a musical environment from a first-person perspective. The game world is simple, and confined, but contains a collection of monolithic objects which the player may interact with. Each of these monolithic objects, when played, adds musical layers to a non-diegetic composition. Gameplay is short, however the experience of playing with these monoliths, in varying combinations and with different intents, allows the player to 'compose' the non-diegetic score to their liking.



Games as Art

Tom Betts 'The Landscape Is The Story'

In the process of producing my current procedural terrain/world engine I've been examining a number of key games that focus on atmospheric terrain as a key design feature. I am a firm believer in notion of landscape as antagonist or exposition. I am generally dismayed at the poor quality of storytelling in most AAA games. I am also infuriated by the general mimicry of hollywood linear cinematic storytelling. Games present a far wider range of channels with which to engage the player in some form of narrative. In my opinion, the space where this interaction takes place is one of the most powerful storytelling mechanisms.

To me, navigation IS narrative and the exploration of game-space proceeds at a tempo guided by a players natural pace rather than the storytelling enforced by linked sets of scripted events. Cutscenes, rambling tutorials and 'story' npcs can ruin immersion, often shaking you out of the game to point out something that the game-world might have been able to tell you itself. Usually Im just reminded of how bad the voice acting, dialogue tree or facial animation is. As a consequence, I find much more empathy and attachment with the subtitled babblings of zeldas charicatures, the disembodied voice of glados or the silent yorda. For me, less is generally more when it comes to character driven exposition, there are a few exceptions but they tend to be swamped in my eyes by the b-movie majority.

Exploration of game worlds is generally an experience conducted on your own terms, to me this makes it an ideal storyteller. I can 're-read' a place, move slowly when I want to take things in or examine the details. Dialogue and cutscenes are usually out of the players control, particularly in the pace of events. Usually the player is locked back into the rollercoaster seat of storytelling, or given a paltry 3 choice menu tree of responses. Back in the game world you are free to explore and experience the 'story' at your own pace, and the world is the paper the story is written on, the frame that holds it, or even the picture that tells it.

In Ico the castle is the enemy, (not in a gothic devil may cry mode), acting with an almost indifferent sense of mystery, history and timeless power. There is no single antagonist, no true conclusion and no dialogue. The only verbal exchanges are the boys plaintive cries to yorda and the brief undecipherable demands of the castles denizens (even these have unreadable glyphed subtitles). The scale of the castle's story is made more epic

through its silence. Where other games would have plaques to read, notes to collect and npcs to interrogate Ico has nothing.

In some ways this is just classic video gaming, from a time before there was space or need for expansive stories. Enemies were wordless opponents, game worlds were abstract spaces that only existed ‘for the player’, even the brief synopsis on the back of spectrum cassette tapes bore little relevance to the game itself. But now many games (and gamers) perceive quantity as quality, and so we are treated to complex stats, achievements, filmic cutscenes, hybrid gameplay etc. *Ico* is haunting, and haunting is something that is very difficult to do in the noise of feature creep.

Shadow of the Colossus revisits this idea of the game-world as antagonist, but as you play you begin to wonder who is the innocent and who is the wrongdoer. The sense of being an interloper is as tangible as it was in Ico and the world is even more melancholy and unknown. Perhaps not as solid now as Ico might be in gameplay terms, some of Sotc seems awkward. The flaws it had on its launch; sluggish controls, low stuttering frame rate and imperfect gameplay are even more obvious when set alongside next gen productions. But there are still no other commercial games like it, the daring contrast of the huge, empty landscape and the gigantic lumbering colossi is remarkable. There is something wrong about it, but then there is something wrong about what the player is being asked to do. The symbiotic nature of the colossi and their domain is effective and intriguing, they are part of the land, part of the game world, part of the story. Sadly there is more actual dialogue (guidance from the temple god) in SotC than Ico but it is still super-minimal and quickly replaced by haunting miles of quiet melancholic ambience.

Another series, another platform. The STALKER franchise is one that initially looks like so many other post apocalyptic FPS fragfests, but its not. The open world gameplay, the brusqueness of the narrative and the strangeness of the world itself combine to tell a story that is more the narrative of a place than of people. Undoubtably indebted to both Roadside Picnic and Tarkovskys *Stalker*, GSC made the scarred and twisted terrain of ‘the zone’ the star ‘character’ of the series. The enemies you encounter, mutant or human, are the products of the zone, twisted genetically and morally through exposure to the menacing alien landscape. The terrain has laid waste to its inhabitants, not the other way around.

Punctuating this land of derelict industrial units and half sunk barges are the anomalies, hazardous regions of concentrated gases, radioactive emissions, electromagnetic forces and toxic liquids. Mechanics-wise they aren’t too disimilar from lava pools or spiked pits, but they have a vital role in personifying the terrain, each location forming an expression of the zones abstract malice and alienation. These locations; the brainscorcher, the oasis, the wish granter, play clearer roles in the story of the zone than the story npcs . The world of Stalker is more populous than Ico or Sotc, but the terse inhabitants converse in downbeat monotones and rarely have much of any detail to say. Missions are usually ethically vague and npcs often act

in exhausted or zombielike states. Their lack of communication and lack of control forces more of the story into the terrain itself.

There are other notable cases of this style of world-story. The Metroid franchise has repeatedly projected the story of dead civilisations (alien & mythical) through its intricate level design, again building a tangible sense of substance through the architectural remnants of absent alien cultures. Even some of the Tomb Raider games follow this archaeological approach to story, albeit with an increasing hollywood edge. But world stories dont need to be macabre or morose, games like oblivion, fallout, silent hill and red dead redemption clearly attempt to mirror their storylines within the landscape. Indeed, all games should do this to some degree, its the basis of anthropomorphic fallacy. But they often mar the experience with poor exposition from other aspects of the gameplay (see my thoughts on Fallout 3).

What is interesting from my perspective and from my current research is whether a world can be generated procedurally and still maintain the ability to speak to players in the way that the above do.

Bibliography

Aallouche, Khalid. Albeiriss, Homam. Zarghoune, Redouane. Arrasvuori, Juha. Eronen, Antti. Holm, Jukka. 'Implementation and evaluation of a background music reactive game', In *Proceedings of the 4th Australasian conference on Interactive entertainment (IE '07)*, Martin Gibbs and Yusuf Pisan (eds.), RMIT University, Melbourne, Australia, Article 1.

Bridgett, Rob. *From The Shadows Of Film Sound*. R. Bridgett self-published, Vancouver, 2010.

Cheng, William. *Sound Play*. Oxford University Press, Oxford, 2014.

Chion, Michel. *Audio-vision: Sound On Screen*. Columbia University Press, New York, 1994.

Collins, Karen. 'An Introduction To The Participatory And Non-Linear Aspects Of Video Games Audio', *Gamesound*. Last updated 2014. Accessed October 2, 2014.
Available: http://www.gamessound.com/interactive.pdf

Collins, Karen. *From Pac-Man To Pop Music*. Ashgate, Aldershot, Hampshire, England, 2008.

Collins, Karen. *Game Sound: An Introduction to the History, Theory, and Practice of Video Game Music and Sound Design*. MIT Press, Massachusetts, 2008.

Collins, Karen. *Playing with Sound: A Theory of Interacting with Sound and Music in Video Games*. MIT Press, Massachusetts, 2013.

Collins, Karen. Kapralos, Bill. Tessler, Holly. *The Oxford Handbook Of Interactive Audio*. Oxford University Press, Oxford, 2014.

Donnelly, K. J. Gibbons, William. Lerner, Neil. *Music In Video Games*. Routledge Music and Screen Media, 2014.

Douglas, J. Yellowlees. Hargadon, Andrew. 'The Pleasures Of Immersion And Interaction: Schemas, Scripts And The Fifth Business'. In *First Person: New Media As Story, Performance And Game*, Noah Wardrip-Fruin and Pat Harrigan (eds.), 1st edition, MIT Press, Cambridge, Mass., 2004.

Ermi, Laura. Mäyrä, Frans. 'Fundamental Components Of The Gameplay Experience: Analysing Immersion'. In *Worlds In Play: International Perspectives On Digital Games Research*, S. De Castell, J. Jenson (eds), 1st ed., Peter Lang Publishing, New York, 2007.

Grimshaw, Mark. *The Acoustic Ecology Of The First-Person Shooter*. VDM Verlag, Saarbrücken, Germany, 2008.

Huiberts, Sander. 'Captivating Sound: The Role Of Audio For Immersion In Games'. Ph.D Thesis, University of Portsmouth and Utrecht School of the Arts, 2014.

Jørgensen, Kristine. *A Comprehensive Study Of Sound In Computer Games*. Edwin Mellen Press, Lewiston, 2009.

Jørgensen, Kristine. 'On The Functional Aspects Of Computer Game Audio'. *Bergen Open Research Archive*. Last updated 2014. Accessed October 2, 2014.
Available: https://bora.uib.no/bitstream/handle/1956/6734/paper-KJorgensen.pdf?sequence=1.

Juul, Jesper. *Half-Real*. MIT Press, Cambridge, Massachusetts, 2011.

Juul, Jesper. 'A Clash Between Game And Narrative'. MBA Thesis. Last updated 2004. Accessed October 30, 2014.
Available: http://www.jesperjuul.net/thesis/.

Kohlrausch, Armin. van de Par, Steven. 'Audio—Visual Interaction In The Context Of Multi-Media Applications'. In *Communication Acoustics*, J. Blauert (eds), 1st ed. Springer, 2005.

Lipscomb, Scott, and Sean Zehnder. 'Immersion In The Virtual Environment: The Effect Of A Musical Score On The Video Gaming Experience'. *Journal of Physiological Anthropology and Applied Human Science,* vol. 23, no. 6, 2004.

Marks, Aaron. *The Complete Guide To Game Audio*. Focal Press/Elsevier, Burlington, Mass., 2009.

Schafer, R. Murray., *The Soundscape: Our Sonic Environment And The Tuning Of The World*. Destiny Books, Rochester, Vermont, 1993.

Stockburger, Axel. 'The Game Environment From An Auditive Perspective'. *Axel Stockburger Homepage*. Last updated 2003. Accessed October 2, 2014.
Available: http://www.stockburger.at/files/2010/04/gameinvironment_stockburger1.pdf.

Stockburger, Axel. 'The Rendered Arena: Modalities Of Space In Computer Games'. *Axel Stockburger Homepage*. Last updated 2006. Accessed November 10, 2014.
Available: http://www.stockburger.at/files/2010/04/Stockburger_Phd.pdf.

Discography

Voigt, Wolfgang (GAS). *Gas*, Mille Plateux, 1996.

Voigt, Wolfgang (GAS). *Zauberberg*, Mille Plateux, 1997.

Voigt, Wolfgang (GAS). *Konigsforst*, Mille Plateux, 1998.

Voigt, Wolfgang (GAS). *Pop*, Mille Plateux, 2000.

Voices From The Lake (Donato Dozzy, Neel). *Voices From The Lake*, Prologue, 2012.

Ludography

140 (2013, Carlsen Games)

Audiosurf (2008, Invisible Handlebar, Dylan Fitterer)

Braid (2008, Jonathan Blow)

DayZ (2013, Bohemia Interactive, Dean Hall)

Fract OSC (2014, Phosfiend Systems)

Half-Life 2 (2004, Valve, Sierra Entertainment)

Papa Sangre (2013, Somethin' Else)

Proteus (2013, Ed Key, David Kannaga)

Resident Evil 4 (2005, Capcom Production, Studio 4, Capcom)

The Bends (2015, svblm)

The Nightjar (2011, Somethin' Else)

The Stanley Parable (2013, Galactic Cafe)

Appendix

Audio-visual material for *Symbiosis*

A full catalogue of audio-visual examples for *Symbiosis* is available at: http://www.andrewhair.co.uk/#!symbiosis/c87r

AV Game Examples

1. *DayZ* - Heightened attention to auditory environments.
Available: https://www.youtube.com/watch?v=SBIRyzER18Y

2. *140* - Second boss.
Available: https://www.youtube.com/watch?v=6mzriunguAE

3. *140* - Music and level progression.
Available: https://www.youtube.com/watch?v=vZou6EaRF2E

4. *140* - Final boss.
Available: https://www.youtube.com/watch?v=pgeUKvMvNaQ

5. *Hotline Miami* - Kinesonic synchresis involving soundtrack.
Available: https://www.youtube.com/watch?v=jawvheSngwE

6. *The Stanley Parable* - Audio narration.
Available: https://www.youtube.com/watch?v=kCB73i5WE2k

Project Diary

- The primary method through which I have been tracing the chronological progression of *Symbiosis* is through GitHub entries. GitHub acts as a cloud storage depot onto which you can periodically submit updates to your project files. All changes to my project have been logged onto GitHub, and can be revisited at any time. The link to my GitHub project is: https://github.com/AndyHairAudio/Symbiosis_Prototyping_01/tree/PC-Local

- As follows are short descriptors for the individual entries that trace back to the beginning of this project.

November

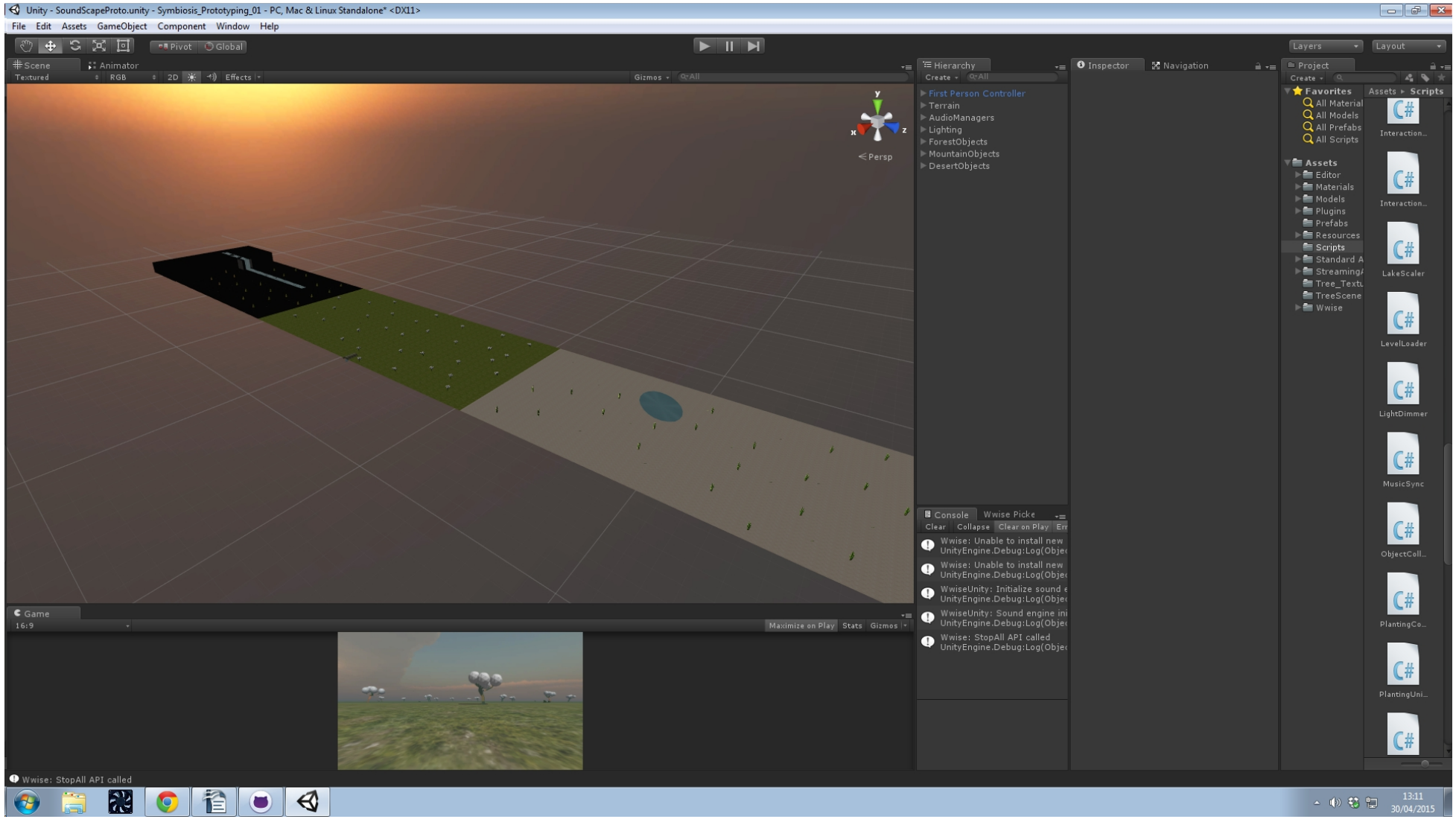
- After visiting my parents in Lincolnshire for a short while, I found that they had cut down a number of tall trees from the garden. Having grown up playing in this garden, there was a strangely acute sense of loss towards these trees. As if they had grown up and watched over me as a youngster while I climbed their branches and played football around their trunks. What was most striking to me was how much more empty the sounds of the garden had suddenly become. Being such a quiet area of countryside, there is very little noise pollution and you can hear for literally miles all around. The significant lack of tree sounds was quite startling. Raises questions as to how strongly we can understand the environment around us just off of the sounds that we encounter.

December

- Beginning experiments with Wwise sound engine. The sound design tools seem incredibly useful for creating ambiences and environments with great granularity. For example I have created a set of insect and creature noises that may be found in a forest environment. The individual samples for these noises can be loaded into Random Containers, which can play continuously, constantly selecting at random new noises to play. My experiments with this focused around creating a day/night cycle of a forest ambience. During the day, there are sounds of winds, leaves rustling, branches creaking, wood breaking, bird song, and insect buzz. During the night, it transitions into the sounds of far-away wolf howls, owl hooting, and cricket chirps. Each of these elements are designed on an extremely granular level – meaning each cricket chirp or other noise is a single sample. Through stacking Random and Blend containers, I have been able to create a constantly evolving ambient environment.
- Starting to try out code snippets to model the behaviour and interactions of individual organisms. Have set up a tree script which grows depending on the availability of water in the area. Finding it very difficult to factor in player interaction into these models however. The goal is to have a naturally balanced system, so organisms have a 'default' state of growth when there is no player interaction. When a player chooses to interact in some fashion, this balance needs to be tipped one way or the other, which can then be translated into musical effects in the sound engine. Engineering this kind of self-sustaining ecosystem which also can be influenced by the player is proving difficult.
- Learning more about C# coding and Unity. Finite State Machines seem extremely useful for what I need to do. FSMs are code blocks which contain discrete code snippets which only run when an object is in a certain state. Using this to dictate the 'health' and growth of trees in the ecosystem and how they play music. E.g. 'Sapling' state plays gentle flute tones, 'Small Tree' state plays violin melodies, 'Big Tree' state plays bowed double bass drones. Tree transitions between these states as it is growing.

January

- Have devised a list of 'zones' or levels which I would like to add in, along with their potential musical elements. Each of these zones have distinct atmospheres, e.g. Forest zone has more woodwind, stringed instruments and keys, as if to sound like a lush, organic ecosystem. Desert zone focuses more on percussion and 'dry' sounding instrumentation. Question of the day: what does a cactus sound like? A simple answer might be that it sounds like a rainstick (which are traditionally built from dried-out cactus skins). But this isn't very interesting. A further question: What does a cactus sound like when it is healthy and growing? When it is dying?
- Beginning to consider how the player could be interacting with this environment. Trying to consider this from the perspective of ecological affects. How do our simple actions of consuming food/water, or gathering wood, affect the ecosystem and how could this be modelled in a game environment? Difficult questions here as our actions have both an immediate, obvious effect, but also a less obvious indirect influence. E.g. we eat the fruit and this grants us a short-term satiation of hunger. This is an immediate, obvious effect on ourselves. However the action of removing the fruit from the ecosystem has less obvious effects on other organisms. The knock-on effect (albeit exaggerated in this model) could be that the lack of that fruit could cause a bird, which relies on such fruit, to die of starvation. This seems to get to the heart of the idea I'm trying to convey through the conduit of sound in a game – how can the convoluted, unseen affects of our most basic choices be conveyed in a more immediate manner?
- Have created a very basic prototype of the 'zones' I'm trying to create, complete with simple organisms. No interaction involved, however each organism has it's own musical signature. Trialling how these individual instruments might sound as a 'complete' ecosystem (orchestra?).


Unity - SoundScapeProto.unity - Symbiosis_Prototyping_01 - PC, Mac & Linux Standalone* <DX11>
File Edit Assets GameObject Component Window Help
Pivot Global
Scene
Animator
Textured RGB 2D Effects Gizmos
Persp
Hierarchy
Create
First Person Controller
Terrain
AudioManagers
Lighting
ForestObjects
MountainObjects
DesertObjects
Inspector
Navigation
Layers
Layout
Project
Create
Favorites
All Material
All Models
All Prefabs
All Scripts
Assets
Editor
Materials
Models
Plugins
Prefabs
Resources
Scripts
Standard A
Streaming
Tree_Textu
TreeScene
Wwise
Assets ▸ Scripts
Interaction...
Interaction...
LakeScaler
LevelLoader
LightDimmer
MusicSync
ObjectColl...
PlantingCo...
PlantingUni...
Console
Wwise Picke
Clear Collapse Clear on Play
Wwise: Unable to install new
UnityEngine.Debug:Log(Obje
Wwise: Unable to install new
UnityEngine.Debug:Log(Obje
WwiseUnity: Initialize sound
UnityEngine.Debug:Log(Obje
WwiseUnity: Sound engine ini
UnityEngine.Debug:Log(Obje
Wwise: StopAll API called
UnityEngine.Debug:Log(Obje
Game
16:9
Maximize on Play Stats Gizmos
Wwise: StopAll API called
13:11
30/04/2015

February

Symbiosis Design Document

Basic mechanics:

- The player has a 'health' system (based upon energy) ranging from 0 to 100.

    - Below 50% energy, the player will start experiencing a distorted reality, including visual and aural hallucinations.
    - The player has a 'regeneration pod' into which they may deposit an amount of energy. Each minute, this pod will regenerate energy at a rate of three times the amount deposited. The player may siphon energy from this pod whenever they wish.

- There are three zones within the game world; a forest, a glacier leading to a river, and a desert plain.
    - Each of these zones have an overarching health system of their own, also ranging from 0 to 100.
        - Within each of these zones are several interactive objects. These objects act as 'energy collectors', into which the player can deposit some of their stored energy.
    - The health level of each of these zones determines the prevalence of creatures, plantlife, water, and weather.
        - Each of these lifeforms carry individual health systems, determined by the abundance of energy in their respective zones.
        - The health of these individual systems is communicated to the player through adaptive music. Examples of this:

Zone	Creature/Lifeform	Musical Elements (Acoustic)	Ambient Elements (Digital)
Forest	Tree	Sapling - Small, wind-like instrumental tones. Small flutes or similar. Bush - Slightly larger, more resonant wind instrument notes. Panpipes etc. Small Tree - Flutes or similar sized wooden wind instruments. Medium Tree - Clarinets? Some fairly sizable wind instrument chords.	Tree Rustling Gentle Winds Forest Floor Noises (Twig/branch cracking, breaking) Bird Flutters Insect Chatter

		Large Tree - Large wind instruments, oboe/digeridoo drones.	
	Bird	Young - Small chimes and whistles. Adult - Medium sized chimes, bell tones or similar.	
	Mammal	Young - Gentle accordion/bagpiped/similar tones. Bright, quiet. Replicative of gentle breathing. Relatively rhythmic, energetic. Adolescent - Slightly slower in rhythm, bassier, slightly louder. Adult - Slow rhythmic drones, bass-heavy. !!Or maybe double bass/cello bowing!!	
Glacier/River	Waterfall	Light Flow - Sparse metallic chimes. Heavy Flow - Dense chimes with ride cymbals.	Howling Winds Birds of Prey Wolf/Lion/Goat Calls
	Fish (salmon)	Small Shoal - Clavier notes. High in pitch, bright and gently played. Dense Shoal - Chaotic clavier notes. A dense, noisy composition.	
	Pine	Double bass.	

	Crow	Young - Slight distortion on electric guitar. Adult - Heavy distortion on guitar.	
Desert	Cactus	Rain stick.	
	Snake	Young - Gentle maraca rolls. Adult - Sustained maraca rolls.	

Additional notes:

- All sounds fall on a spectrum between sparseness/dissonance, and richness/harmony. This should be linked to how prosperous the ecosystem is at any one time. E.g. Widespread growth produces more harmonious soundscapes, decay produces harshness.
- Day and night cycles should be possible.
- Weather systems may be included, subject to time expenditure and deadline limitations.
- Subzones could include caves, oases, and pastures.
- There should be game parameters that influence the sound of each and every environment on different scales. For example:

Scale	Game Parameter	Sonic Effect
Global	Time of Day (24hr)	Ambient noise volume should increase during the day, as should the density of wildlife elements. Individual elements could be modulated during nighttime. For example the musical trees could trigger less frequently, decrease in pitch, etc.

March

- Encountered a major pitfall with the idea of using game-world organisms as spatialized instruments. The Wwise engine contains a tool dedicated to interactive music, which keeps a 'global tempo' between segments and tracks. This allows for dynamic changes in music to occur in a proper fashion. However, when attempting to apply these segments to multiple game objects in the 3D game world (and as such have attenuations), the music does not keep in time. This seems to be a limitation in the technology of game audio engines as they currently stand. In a sense I am attempting to work in the grey area between conventional sound design (spatial, applied to objects in diegesis, attenuated, without time signature) and formalised score/music (with a time signature/key, typically non-diegetic, outside of 3D game space). A way to look at this would be to have a few radios playing music in the game space. Each individual radio is capable of keeping time with the music that is playing from itself. They can transition between music tracks in time and key within themselves. However if one were to stand in the space between these radios, where music can be heard from all, then the music cannot be kept in time. Have to reconsider this approach.
- As a result of the difficulties encountered with spatial instrument tracks I've decided to approach the music side of Symbiosis as a non-diegetic score. This score is still to be reactive to player interactions and the effects that they have, but will no longer be applied to objects in the game world. To do this I've considered the score from three perspectives:

    - Player Events

        Event: Player picks up a fruit
        Meaning/Signifier: Opportunity

        Event: Player Plants a seed
        Meaning/Signifier: Positive Reinforcement (Environmental)

        Event: Player Eats a Seed
        Meaning/Signifier: Positive Reinforcement (Individual)

Event: Seed Created By Tree

Meaning/Signifier: Draw Attention To and Suggest Life

– Musical/Sound Design Motifs (Narrative 'Triggers')

- Entering the world
- Discovering a tree
- Discovering the lake
- Finding a seed
- Picking up a seed
- Eating a seed
- Planting a tree

- Reactive Music Motifs (Played as Non-diegetic score)

– Bird Theme (Piano keys, growing in density as the bird population grows)
– Fish Theme (Stringed instrument bows, gentle at first, but with more intense rhythms as the lake becomes populous)
– Player planting trees (Percussive drum rhythms, playing when the player starts planting and developing as they succeed in planting more trees)
– Player fishing (Percussive drum rhythms, lower tempo than the planting of trees)

April

- Visual design is largely complete. Gameplay is still in progress. Have discovered a way to have the sound engine send messages regarding tempo/key etc back into Unity. This presents an opportunity to use the music itself as a way for the player to interact. Have started creating short minigames, when the player starts fishing or planting trees, where the player has to time their actions with the music in order to be successful. My intent is for this to reinforce the idea in the player that the game world is governed by music and it's character (and vice-versa). As a result I have had to revise some of the composition I had been doing to suit this new approach. The music now has to have a strong rhythmic element so the player has a 'guide' to help them play effectively.
- Through experimenting with these music messages from the sound engine, I have discovered a way to have the spatialized instruments that I had originally wanted. Although it is a bit late to revise the whole project to use this approach, I am attempting to add in a few creatures that respond to music in this way. For example, the birds now periodically sing out with synthesizer motifs as they fly overhead, which keeps in time with the music. This is proving a very rewarding discovery, and I feel I am breaching uncharted territory in terms of how games have used responsive, evolving music.
- Have had a few sessions of user-testing with second-year students to gauge how players respond. Discovered that many like the ideas and interactions in the project, however they feel as though they missed a large portion of the gameplay/music due to the lack of explicit direction. As a result I have added in several 'monoliths' into the game world, which have a unique musical signature. The monoliths sound out for a considerable distance in the game world, so as to draw attention to themselves and encourage the player to approach. The player can read these monoliths, and will be given short excerpts of narrative script, along with brief instructions on how they may interact with the world.
- Major bug issues are starting to arise. Significantly, there are latency issues with the digging and fishing interactions. The sequences do not always pop up at right time on the beat. They lag a little bit behind. Additionally, the game has been designed to accommodate long-term installation in an exhibition. Therefore the game has to be able to transition from the main menu ^ into the level ^ back to the menu repeatedly over the course of a day. This is presenting issues with sound as when the player re-enters the game, there is a 'ghost' of sound from where the previous player left off. This may be a simple fix, or it may require a regular restarting of the game. After discussing with

professional Wwise developers on the Facebook group, I have found a way to stop all outgoing sound when the level is reloaded. As a result the player only experiences the sound for their current playthrough.