

Bachelorarbeit

Hochschule der Medien Stuttgart

Fakultät Electronic Media

Studiengang Audiovisuelle Medien (AM7)

From Sound Source to In-Game Sound

Ein Leitfaden zur Vertonung von Computerspielen am
Beispiel eines Survival-Games

Vorgelegt von: Michael Wagner (28932)

Vorgelegt am: 31. August 2017

Erstprüfer: Prof. Oliver Curdt

Zweitprüfer: Prof. Dipl.-Ing. Uwe Schulz

Eidesstattliche Erklärung

Hiermit versichere ich, Michael Wagner, ehrenwörtlich, dass ich die vorliegende Bachelorarbeit mit dem Titel: „From Sound Source to In-Game Sound - Ein Leitfaden zur Vertonung von Computerspielen am Beispiel eines Survival-Games“ selbstständig und ohne fremde Hilfe verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen wurden, sind in jedem Fall unter Angabe der Quelle kenntlich gemacht. Die Arbeit ist noch nicht veröffentlicht oder in anderer Form als Prüfungsleistung vorgelegt worden.

Ich habe die Bedeutung der ehrenwörtlichen Versicherung und die prüfungsrechtlichen Folgen (§26 Abs. 2 Bachelor-SPO (6 Semester), § 24 Abs. 2 Bachelor-SPO (7 Semester), § 23 Abs. 2 Master-SPO (3 Semester) bzw. § 19 Abs. 2 Master-SPO (4 Semester und berufsbegleitend) der HdM) einer unrichtigen oder unvollständigen ehrenwörtlichen Versicherung zur Kenntnis genommen.

Stuttgart, den 31. August 2017

Michael Wagner

Kurzfassung

Der Bereich der Computerspiele ist geprägt von rapide aufeinanderfolgenden Neuerungen; die immer schnelleren Prozessoren und leistungsfähigeren Grafikkarten, Innovationen im Gebiet der Simulationen (wie zum Beispiel 3D-Audio-Verfahren) und der Software. Sie alle befeuern die rasanten Entwicklungssprünge im Bereich der Games.

Doch auch abseits der technischen Möglichkeiten gibt es immer wieder neue Erfindungen im kreativen Bereich: stets werden neue Spielmechaniken erfunden, weiterentwickelt und mit bereits bestehenden, in neuartiger, kreativer Weise, verknüpft.

In dieser Arbeit soll das Survival-Genre genauer betrachtet werden, das aufgrund seiner Spielmechaniken ganz besondere Herausforderungen in der Vertonung birgt. Die Erkenntnisse können weiterführend – je nach Genre – auch auf Produktionen von anderen Spielen angewandt werden und sollen als Leitfaden verstanden werden.

Dazu wird nicht nur der reine Prozess der Computerspielvertonung, von der Vorproduktion bis zur Implementierung in einer Middleware beleuchtet, sondern darüber hinaus auch eine fundierte theoretische Grundlage geschaffen, welche die Planungsphase vereinfacht und die Wirkungsweise von Sound Design in Spielen erklärt, um so den Prozess der Vertonung zu optimieren.

Abstract

The field of video games is heavily affected by fast innovation; with processors getting faster and faster, more powerful graphics cards, innovations in the field of simulation (e.g. 3D-audio techniques) and software, video games are taking a great leap forward.

Not only technology takes big leaps forwards, but the creative sector as well: It seems that every day new game mechanics are invented, improved or creatively combined with pre-existing ones.

This thesis focuses on the survival genre which has a special set of challenges in terms of audio production, due to its game mechanics. The conclusions of this thesis can be translated to other game productions – depending on the genre. It is intended as a guide for video game sound.

Therefore, not only the process of getting audio into a video game, from pre-production to implementation within a middleware, are discussed. A solid theoretical foundation is built which will help to simplify the process of planning, clarify the effectiveness of sound design in games and will further optimize the process of game audio.

Inhaltsverzeichnis

Eidesstattliche Erklärung.....	2
Kurzfassung.....	3
Abstract.....	4
Inhaltsverzeichnis.....	5
Abbildungsverzeichnis.....	7
Abkürzungsverzeichnis	8
Vorwort.....	9
1 Vertonung von Filmen und Games im Vergleich.....	10
1.1 Nonlinearität und Interaktivität.....	10
1.2 Spotting.....	13
1.3 Sound Kreation	14
1.4 Mixing.....	14
1.5 Teamgrößen und Aufgabenverteilung	14
2 Computerspielgenres	16
2.1 Vorarbeit: Übergeordnete Genres des Survival	16
2.1.1 Action	17
2.1.2 Adventure	17
2.1.3 Action-Adventure	18
2.1.4 RPGs	18
2.2 Das Survival-Genre	19
3 Vor der Vorproduktion: Theorie	21
3.1 Mögliche Rollen von Game-Audio.....	21
3.1.1 Immersion.....	22
3.1.2 Aurales Feedback.....	24
3.1.3 Auditive Identifikation.....	27
3.1.4 Räumliche Situierung	29
3.1.5 Gameplay Identifikation	30
3.1.6 Stille.....	30
3.2 Dynamische Vertonung	31
3.3 Diegese in Bezug auf Games	32
3.4 Rolle von Stereotypen in der Vertonung	33
4 Die Produktion	36
4.1 Vorproduktion.....	36

4.1.1	Kommunikation mit dem Team: Producer und Game-Designer	36
4.1.2	Kommunikation mit dem Team: Art-Department	39
4.1.3	Kommunikation mit dem Team: Programmierung.....	39
4.1.4	Technische Rahmenbedingungen	42
4.1.5	Was benötigt alles eine Vertonung?	44
4.1.6	Akustische Perspektive	46
4.2	Sound Design.....	49
4.2.1	Varianz und Abnutzung.....	49
4.2.2	Sound Libraries.....	53
4.2.3	Studioaufnahmen	54
4.2.4	Field-Recording	56
4.2.5	Nachbearbeitung.....	58
4.3	Exkurs – Voice Over und Voice Acting	68
4.4	Exkurs – Musik.....	69
5	Middleware.....	71
5.1	FMOD Studio	72
5.1.1	Audio Bin.....	72
5.1.2	Events	73
5.1.3	Soundmodule	75
5.1.4	Randomisierung.....	79
5.1.5	Triggerverhalten	79
5.1.6	Parameter	80
5.1.7	Sound-Assets in Parametern	81
5.1.8	Echtzeit-Layering	82
5.1.9	Logic Track.....	85
5.1.10	Begrenzung der Anzahl getriggelter Events	86
5.1.11	Snapshots	87
5.1.12	Soundbanks.....	90
5.1.13	Live Update	91
5.2	Abschluss der Vertonung.....	92
6	Fazit.....	94
	Verweise.....	95

Abbildungsverzeichnis

Sofern nicht anders angegeben, handelt es sich bei den Abbildungen um eigens erstellte Grafiken und Screenshots.

Abbildung 1: HUD aus THE DARK SIDE OF THE MOON (Fleischmann, 2017)	25
Abbildung 2: Zugriff der UE4 auf FMOD, Ausgangssituation	41
Abbildung 3: Zugriff der UE4 auf FMOD, Fehler	41
Abbildung 4: Zugriff der UE4 auf FMOD, Korrigierte Version.....	41
Abbildung 5: Spielszene aus THE WITCHER 3: WILD HUNT (Thudium, 2015).....	48
Abbildung 6: De-Hum Modul aus iZotope RX 6.....	59
Abbildung 7: Spektrogramm und Wellenform aus iZotope RX 6	60
Abbildung 8: Spektrogramm aus Sequoia 14 mit deutlich erkennbarem Klick.....	61
Abbildung 9: statisches Layering	62
Abbildung 10: Loop-Erstellung, das Ausgangsmaterial	65
Abbildung 11: Loop-Erstellung, Schnitt an einem Nulldurchgang.....	65
Abbildung 12: Loop-Erstellung, Tauschen der Clips/Objekte.....	65
Abbildung 13: Loop-Erstellung, Crossfade anlegen	66
Abbildung 14: Objektbasiertes Arbeiten im Pre-Mix	67
Abbildung 15: FMODs Audio Bin.....	73
Abbildung 16: 3D Panner	74
Abbildung 17: Stereo Panner links, Mono rechts (kein Panner).....	74
Abbildung 18: Trigger Region	75
Abbildung 19: Single Sound	76
Abbildung 20: Multi Sound	77
Abbildung 21: Scatterer Sound	78
Abbildung 22: Event Sound	78
Abbildung 23: Randomisierung	79
Abbildung 24: Trigger Behavior	80
Abbildung 25: Hinzufügen von Parametern	80
Abbildung 26: Mehrere Parameter in FMOD	81
Abbildung 27: Sound Assets in einem Parameter, Lockpicking.....	82
Abbildung 28: Fußschritte auf hartem Untergrund	83
Abbildung 29: Parameterautomation	83
Abbildung 30: Parameter im Triggerverhalten	84
Abbildung 31: Logic Track und Marker	85
Abbildung 32: Feuersound aus THE DARK SIDE OF THE MOON	86
Abbildung 33: Event Macros	87
Abbildung 34: Routing der Charaktersounds.....	88
Abbildung 35: Ducking via Snapshot	89
Abbildung 36: Hallraum via Snapshot.....	90
Abbildung 37: Build Preferences	91

Abkürzungsverzeichnis

ADR	Automated dialogue replacement
AHDSR	Attack, Hold, Decay, Sustain, Release
AI	Artificial Intelligence
DAW	Digital Audio Workstation
dB	Dezibel
dB FS	Decibels relative to full scale
FX	Effects
GDD	Game Design Document
HUD	Head-Up-Display
KI	künstliche Intelligenz (vgl. AI)
Medikit	Medical Kit
Mod	Modification
NPC	Non-Player Character
O-Ton	Original Ton (Set Ton)
PCM	Pulse-Code-Modulation
RPG	Role Playing Game
RTS	Real-Time Strategy
TBS	Turn-Based-Strategy

Vorwort

Die Produktion eines Computerspiels ist eine äußerst interdisziplinäre Angelegenheit. Dennoch kommt es immer wieder vor, dass der Vertonung zu spät gedacht wird. Kein Wunder; oft wird sich am Workflow einer Filmproduktion orientiert, die jedoch in einem völlig verschiedenen Zeitplan abläuft. Da darunter die gesamte Spielerfahrung leiden und das Potential eines Games nicht vollständig entfaltet werden kann, gilt es dies zu vermeiden.

Dieser Leitfaden soll die, für den Tonschaffenden wichtigen Schritte einer Computerspielproduktion erörtern. Doch zunächst sollen die bereits angedeuteten Unterschiede von Film- und Computerspielvertonung geklärt werden, um damit zu verdeutlichen, dass, und vor allem warum, verschiedene Herangehensweisen vonnöten sind.

Dabei werden Einblicke in die Vertonung von THE DARK SIDE OF THE MOON, einer Game-Studioproduktion Interaktive Medien des Sommersemesters 2017 gewährt. Da es sich bei dieser Produktion um ein Survival-Spiel handelt, sollen auch das Genre und seine Besonderheiten genauer untersucht werden. Wie später noch deutlich wird, nehmen das Genre und die jeweilige Stilrichtung innerhalb dieses Genres, einen starken Einfluss auf die Vertonung. Im Zuge dessen wird erörtert, was ein Genre im Falle von Computerspielen ausmacht.

Es soll anschließend eine theoretische Grundlage von Vertonung allgemein und in Bezug auf Computerspiele geschaffen werden, die dabei helfen soll, die Wirkungsweise von Sounds in Computerspielen zu verstehen. Durch dieses Verständnis lassen sich die einzelnen Schritte der Tonarbeit, insbesondere die Vorproduktion, effizienter gestalten.

Es folgen die einzelnen Produktionsschritte, die aus einer praxisorientierten Sicht begleitet und mit Beispielen untermalt werden sollen. Es soll geklärt werden, was alles in einem Game einer Vertonung bedarf und welche Dokumente oder Informationen vom Produktionsteam eingeholt und abgeklärt werden müssen. Im Zuge des Sound Designs werden auch das Thema der Varianz, sowie Abnutzung von Sounds diskutiert und darüber hinaus Wege vorgestellt, wie neue Sounds kreiert werden können. Ein kurzer Exkurs zu Voice Over und Musik in Computerspielen findet ebenfalls statt; das Hauptaugenmerk der Arbeit ist jedoch das Sound Design.

Die Implementierung der Sounds innerhalb einer Middleware wird am Beispiel von FMOD besprochen. Sie beinhaltet unter anderem das Anlegen von Events und Banks und behandelt die unterschiedlichen Sound Module.

Der besseren Lesbarkeit halber wird auf die Nennung beider Geschlechterformen abgesehen. Insofern eine Personen-, Gruppen oder Berufsbezeichnung genannt ist, ist stets auch die weibliche Form gemeint.

1 Vertonung von Filmen und Games im Vergleich

Das Wissen über die Vertonung von Filmen ist umfassend, so handelt es sich beim Tonfilm um eine etablierte Medien- und Kunstform. Computerspiele sind, verglichen dazu, eine sehr junge Erscheinung. Während beide Formate viele Parallelen aufweisen – beides sind stark visuell geprägte Medien –, so unterscheiden sie sich doch signifikant voneinander. In diesem Kapitel sollen eben jene Ähnlichkeiten und Unterschiede der Film- und Computerspielvertonung beleuchtet werden, vor allem bezüglich des Workflows.

1.1 Nonlinearität und Interaktivität

Der wohl größte Unterschied zwischen Games und Filmen ist die Nonlinearität und Interaktivität ersterer. Um dies zu verdeutlichen wird nun zuerst Film, der – genau wie Musik – ein lineares Medium darstellt, betrachtet.

Die Vertonung eines Films ist starr an seinen zeitlichen Ablauf, beziehungsweise die Frames gebunden. Der Film zeichnet sich durch seine Vorhersagbarkeit aus: Im finalen Schnitt wird eine Handlung immer an derselben Stelle auftreten. Für den Sound Designer bedeutet dies, dass eine Synchronisation auf Zeitbasis möglich ist (Horowitz & Looney, 2014).

Die Vertonung findet folglich überwiegend, in manchen Fällen auch ausschließlich, in der Postproduktionsphase statt. Hier kann der Schnitt in eine DAW importiert werden und der Sound Designer, beziehungsweise Filmtonemeister, ist dort in der Lage, die diversen Soundeffekte, Dialoge und Musikstücke in einer zeitlichen Reihenfolge anzuordnen, oder sogar diese synchron zum Film aufzunehmen (Beispiele hierfür sind Foley-Effekte¹ und ADR²). Im Anschluss kann ein Mix erstellt werden, der schlussendlich in eine einzelne Datei *gibounced*, also gespeichert wird (Leenders, 2012).

In Computerspielen gibt es, bedingt durch den Handlungsfreiraum des Spielers, diese zeitliche Gewissheit nicht. Sobald auch nur eine einzige Entscheidungsmöglichkeit gegeben ist, ist die Spielerfahrung interaktiv. Diese Interaktivität impliziert einen nonlinearen Ablauf. Nach Leenders (2012) kann Linearität ab dieser Entscheidungsmöglichkeit nicht mehr gegeben sein, da nicht vorhersehbar ist, für welchen Weg sich der Spieler entscheiden wird. Ein Spiel wird folglich, mit zunehmender Zahl an Interaktionsmöglichkeiten, immer komplexer und unvorhersehbarer.

¹ Foley meint klassischerweise zum Bild synchron aufgenommene Charaktergeräusche, wie z.B. Fußschritte und Kleidungsrascheln. Gelegentlich wird der Begriff auf andere Soundeffekte erweitert (Ament, 2009), jedoch wird eigentlich alles was nicht klassisches Foley ist, als Sound Effekt bezeichnet (Leenders, 2012).

² ADR bezeichnet die lippenasynchrone Nachvertonung von Dialogen (Bender & Wulff, 2012).

Dem Spieler steht es frei, wie lange er sich an welcher Stelle der Spielwelt befindet und wann er welche Aktion ausführt. Es ist also nicht möglich, die Vertonung an ein starres Zeitraster zu koppeln, da die Vertonung in Echtzeit auf das Spielgeschehen, die Spielwelt und Einflüsse, wie die Player-Status oder andere Parameter, reagieren kann (Weiss, 2015). Stattdessen werden einzelne Soundeffekte an Events, also Aktionen des Spielers, oder andere Parameter gebunden. Letztlich bedeutet dies, dass jeder Sound als separate Datei vorliegen muss (Horowitz & Looney, 2014).

Es müssen also die Bedingungen für einen Sound sowohl mit den Game Designern, als auch mit dem Audio Programmierer definiert werden. Als Konsequenz daraus ziehen Horowitz und Looney (2014), dass auch jede Änderung am Game Design und den Spielmechaniken, potentiell die Art und Weise wie die Sounds getriggert und beeinflusst werden, stark verändern kann.

Dies ist bei der Möglichkeit der Unterbrechung des Spiels der Fall, einem wichtigen Bestandteil der Non-Linearität von Games (Leenders, 2012). Damit ist nicht nur ein potentieller Pause-Modus gemeint, sondern die Unterbrechungen die sich durch Speichern und Laden, oder aber dem Beenden und zu einem späteren Zeitpunkt laden, ergeben. Die Frage, die sich hier nach Leenders (2012) stellt ist, welche Art des Speicherns und Ladens gibt es?

- a) *Wiedereinstige nur am jeweiligen Level-Anfang*: Besonders bei älteren Spielen ist dieses Vorgehen üblich; hier gibt es zum Beispiel Level-Codes, die man freispielen kann und mit denen man an ein bestimmtes Level springen kann. Ein Beispiel hierfür ist Star Trek: 25th Anniversary (1992, Gameboy). Der Einstieg in ein neues Level, ob mitten im Spielfluss, oder nach dem Wiederaufnehmen unterscheidet sich nicht nennenswert, weshalb er aus Sicht des Tonschaffenden keinen oder nur kaum Mehraufwand darstellt (Leenders, 2012).
- b) *Automatisch angelegte Speicherpunkte*, zum Beispiel beim Betreten spezieller Orte oder Regionen: Diese werden häufig in Spielen bei der Spielstandauswahl als Automatisches Speichern benannt (nicht zu verwechseln mit Quicksave). Oft werden sie in der Ruhe vor dem Sturm oder einer anderen ruhigen Situation angelegt, was beim Wiederaufnehmen des Spielgeschehens und des Etablierens der akustischen Situation hilfreich ist (Leenders, 2012). In THE ELDER SCROLLS V: SKYRIM (2011) ist dies der Fall: Speicherpunkte werden hier etwa beim Betreten von Dungeons oder Häusern angelegt. Ähnlich wie in a) muss ohnehin eine neue Situation etabliert werden. Jedoch besitzt der Spieler hier auch die Möglichkeit von einer anderen Stelle im Spiel diesen Spielstand zu laden, es muss also ein möglicher Übergang des Tons bedacht werden.

- c) *Checkpoints*, beispielsweise vor oder nach schweren Passagen: Ein Checkpoint muss nicht zwingend auch einen Spielstand beinhalten³. In der Regel setzt ein Checkpoint den Spieler direkt wieder in dieselbe Situation im Spiel und das in relativ kurzer Zeit. Eventuell kann dabei die bisherige Musik und Atmo einfach fortgesetzt werden, oder aber, nach einem kurzen auralen Feedback des Spielertodes, wiederaufgenommen werden (Leenders, 2012). Die Gestaltung von Übergängen wird dadurch begünstigt, dass grob bekannt, ist in welcher akustischen Situation sich der Spieler befindet: Gegeben sei ein Level, bestehend aus den Checkpoints A, B und C. Checkpoint A wird nur dann geladen, wenn der Spieler vor dem Erreichen von B stirbt, B nur beim Tod vor dem Erreichen von C und C nur beim Tod vor dem Abschluss des Levels. Was manuell geladen werden kann, hängt davon ab, ob Checkpoint und Speicherpunkte miteinander verknüpft sind; man beginnt entweder am Level-Anfang (Checkpoint ist kein Speicherstand), oder an dem letzten freigeschalteten Checkpoint (Checkpoint und Speicherstand fallen zusammen). Ein Beispiel für diese Vorgehen ist CALL OF DUTY – WORLD AT WAR (2008). Hier sind die einzelnen Levels in Checkpoints unterteilt, die nicht alle frei ladbar sind. Besitzt der Spieler nur noch wenig Lebensenergie, so ist ein Herzklopfen zu hören. Stirbt er, so schlägt das Herz ein letztes Mal, die Umgebungsgeräusche werden mit einem Tiefpass gefiltert und der Ton blendet rasch in Stille aus; der letzte Checkpoint wird geladen.
- d) ‚*Freies Speichern*‘ an Checkpoints: Der Spieler kann hierbei nur an bestimmten Punkten speichern, dafür aber ohne an ein Event oder einen Fortschritt gebunden zu sein. Lediglich der Checkpoint muss zugänglich sein (Leenders, 2012). Es ist also nicht, oder nur schwer abschätzbar, von wo aus der Spieler einen Spielstand, an welchem Ort innerhalb der Spielwelt laden wird. Denkbar ist es, dass der Tonschaffende bei einer überschaubaren Anzahl von vorher bekannten Checkpoints, eigene Übergänge anlegt. Ansonsten kann zum Beispiel auf Stille ausgeblendet und die neue Situation frisch etabliert werden, oder aber die alte und neue Soundumgebung mit einem Crossfade überbrückt werden. Beispiele hierfür sind HORIZON ZERO DAWN (2017) – hier kann das Spiel an entdeckbaren Lagerfeuern gespeichert werden – oder THE FOREST (2014); hier muss der Checkpoint in Form eines Unterschlupfes selbst gebaut werden.
- e) *Freie Speicherstände*: Hiermit ist das freie Speichern ohne Ortsbindung, zu beinahe jedem Zeitpunkt gemeint. Auch eine Quicksave-Funktion fällt unter diese Kategorie. Die-

³ Beendet der Spieler an einem solchen Checkpoint das Spiel, so gehen Spielinhalte verloren und das Spiel wird am letzten richtigen Speicherpunkt geladen.

ses Vorgehen ist besonders bei Rollenspielen beliebt. In diesem Fall hat der Tonschaffende die geringste Kontrolle über den Übergang beim Laden im laufenden Spielgeschehen. Dementsprechend sind Fades auf Stille, oder Crossfades recht häufig anzutreffen (Leenders, 2012).

Weniger problematisch als das Laden im laufenden Betrieb, fällt für die letzten beiden Optionen die Wiederaufnahme des Spiels nach einer Unterbrechung aus, da der Spieler sich nicht in einer anderen Umgebung der Spielwelt und somit noch keiner etablierten Emotion befindet.

1.2 Spotting

Nach Schmidt (2003) wird die Evaluation, an welchen Stellen Sounds oder Musik benötigt werden, als Spotting bezeichnet. Im Film geschieht dies in der Regel nach dem Picture-Lock⁴ oder zumindest auf Basis eines groben Schnittes. Dies bietet dem Sound Designer einiges an Sicherheit: „[...] das Timing der Aktion zumindest innerhalb der Szenen steht fest und auch der gesamte Spannungsbogen und die Visualität des Produktes sind erkennbar.“ (Leenders, 2012, S. 33)

Das Spotting bei Computerspielen findet nicht in der Post-Produktionsphase statt, sondern so früh wie möglich im Produktionsprozess. Dies bedeutet aber auch, dass keine In-Game Sequenzen verwendet werden können. Es muss also anhand von ersten Artworks, Storyboards und allgemeinen Informationen, wie Setting, Plot, Genre und Stimmung, eine Quintessenz gewonnen werden, mit der der Sound Designer arbeiten kann. Erst später im Verlauf der Produktion stehen Bilder und noch später tatsächliches Game Play zur Verfügung (Leenders, 2012). Des Weiteren ist es laut Leenders (2012) nicht unüblich, dass auch drastische Änderungen noch mitten in der Produktion vorgenommen werden. Er nennt als Beispiel TOMB RAIDER (1996); ursprünglich war ein männlicher Protagonist geplant. Dieses Konzept wurde jedoch verworfen und mit Lara Croft eine Ikone der Videospiegelgeschichte geschaffen. Dass diese Entscheidung folgeschwer für die Vertonung ist, sieht man daran, welche Geräusche allein mit dem Geschlecht des Protagonisten zusammenhängen; angefangen von Fußschritten bis hin zu Dialog und Voice FX.

Als Schlussfolgerung lässt sich sagen, dass in Games, anders als im Film, nicht die Frames, sondern die möglichen Events, beziehungsweise Triggers gesichtet werden (Schmidt, 2003). Auf die Kriterien, nach welchen beurteilt wird, was in einem Computerspiel vertont werden muss, wird später näher eingegangen (→ Kapitel 4.1.5).

⁴ Der Picture-Lock ist der Zeitpunkt, ab welchem der Schnitt im Film nichtmehr verändert wird.

1.3 Sound Kreation

Das Erstellen der eigentlichen Sounds für Computerspiele unterscheidet sich, verglichen mit den anderen Aspekten, nur geringfügig von dem für Film. Deshalb soll es an dieser Stelle nur sehr kurz aufgeführt werden. Field-Recording zur Aufnahme von neuen Geräuschen, die Verwendung von Sounds aus Libraries und auch Foley werden verwendet (wobei Foley im Bereich der Computerspiele allgemeiner als Charaktersounds zu verstehen ist, da die bildsynchrone Aufnahme nur im Falle von gerenderten Cutszenes möglich ist) und auch Layering wird eingesetzt. Der zu beachtende Unterschied im Workflow ergibt sich aus der Anforderung, dass alle Sounds einzeln vorliegen müssen (Schmidt, 2003). Einen tieferen Einblick in die Schaffung von Sounds wird in Kapitel Sound Design (→ 4.2) ermöglicht.

1.4 Mixing

Die Vorgehensweise einer Filmmischung kann man vereinfacht wie folgt zusammenfassen: Der Übersichtlichkeit halber und um den letzten Mischschritt zu vereinfachen, werden Pre-Mixe der verschiedenen Departments angelegt (z.B. O-Ton, Sound FX und Musik). Diese werden ge-bounced. Für den *Final Mix* werden diese *Stems* noch einmal untereinander in der Lautstärkere-lation angepasst, sodass sie ein schlüssiges Gesamtklangbild ergeben.

Da in einem Computerspiel nicht festgelegt ist, zu welchem Zeitpunkt welcher Sound spielt, bleibt auch die Frage, welche simultan gehört werden, unbeantwortet. Einen Final Mix, im klassischen Sinne, kann es also nicht geben. Vielmehr ist es so, dass die Audio Engine den Mix in Echtzeit erstellt; die Aufgabe des Sound Designers ist es hierbei also, die Regeln zu definieren, nach welchen dieser Mix stattfindet (Schmidt, 2003).

Es ist jedoch anzumerken, dass der Sound Designer durch die Verwendung von Middleware einiges an Kontrolle zurück gewinnt. Neben weiteren Themen, wird auf den Mixing-Prozess von Computerspielen und auch das sogenannte Live-Mixing im Kapitel 5 eingegangen.

1.5 Teamgrößen und Aufgabenverteilung

Es lässt sich sagen, dass die Audio-Teams in Filmen in der Regel größer sind als im Games-Bereich. Zwar lässt sich bei AAA-Titeln ein Trend hin zu größeren Teams erkennen, jedoch ist in vielen kleineren Produktionen und Entwicklerstudios die Tonarbeit in Personalunion verwirklicht (Rumsey, 2010).

Laut Phillips (2014) verwundert es in diesem Kontext auch nicht, dass die Positionsbezeichnungen und deren Bedeutung von Entwicklerteam zu Entwicklerteam und von Projekt zu Projekt

varyieren können. Im genannten Falle der Personalunion, ist der *Audio Director* nicht nur für die Planung zuständig. Im Umfeld der kleineren Entwicklerstudios übernimmt er, falls überhaupt einer Teil des Teams ist, die Verantwortung für die Vorproduktion (Kommunikation mit den anderen Departments, Spotting, Planung der Aufnahmen, etc.), die Organisation und/oder das eigentliche Sound Design (Aufnahme von Foley, Field-Recording, Voice-Over, Library-Recherche, Layering, etc.), in manchen Fällen für die Musik und letztlich die Implementierung mittels Middleware.

Auch die Bezeichnung *Sound Designer* ist nicht so klar, wie sie auf den ersten Blick scheint: nach Horowitz und Looney (2014) kann mit der Positionsbezeichnung in der Games-Branche sowohl das reine Schaffen von Sounds, als auch zusätzlich die Implementierung dieser beinhalten. Demnach ist die Unterteilung in *Audio Director* und *Sound Designer* in diesem Bereich oft nicht möglich.

Gerade im Bereich der Survival-Games, welche eher unter den Independent-Produktionen angesiedelt scheinen, ist eine solche Situation durchaus wahrscheinlich. Hierzu drei Beispiele von beliebten Survival Games:

1. **Hinterland Studio** aus Kanada, Entwicklerstudio des Art-Survival Games THE LONG DARK (2014)), ist eines der größeren Indie-Studios und besteht aus 22 Mitarbeitern. Einer von ihnen ist Audio Director (Hinterland Studio Inc., kein Datum).
2. **Endnight Games**, ebenfalls ein kanadisches Entwicklerstudio und verantwortlich für das Survival Spiel THE FOREST (2014), besteht aus drei festen Mitgliedern und greift bei Bedarf auf Freelancer zurück. Man kann also sagen, dass das Team hin und wieder eine Größe von bis zu 10 Personen umfasst (Endnight Games, kein Datum).
3. Das Entwicklerteam von STRANDED DEEP (2015), **Beam Team**, besteht aus lediglich zwei Programmierern. Das Studio ist in Australien beheimatet (Beam Team Pty Ltd, kein Datum).

2 Computerspielgenres

Die Einteilung von Computerspielen in Genres findet „[...] nicht anhand von visuellen Kriterien, Erzählweise oder Szenari[en] [...]“ statt (Leenders, 2012, S. 18). Weiter legt Leenders (2012) nahe, dass die Einteilung vielmehr an die Spielmechaniken gekoppelt ist. Ein Beispiel: Solange ein Computerspiel die Mechaniken des Survival-Genres erfüllt, ist es irrelevant, ob es in einer düsteren Zombieapokalypse stattfindet, oder aber Entenküken in einer fröhlichen Märchenwelt als Protagonisten dienen.

Da Spiele häufig verschiedene Spielmechaniken kombinieren und daraus wiederum neue User-Experiences schaffen, ist die Zuordnung von Computerspielen in Genres nicht immer eindeutig, geschweige denn unumstritten⁵ (Leenders, 2012).

Die Arbeit am Computerspielton verschiedener Genres, fällt zum Teil grundlegend unterschiedlich aus (Leenders, 2012). Bei der Vertonung von Computerspielen besteht zudem eine gewisse Erwartungshaltung der Konsumenten (aber auch des Entwickler-Teams) dem Ton gegenüber; diese gehen mit dem jeweiligen Genre des Spiels einher. Dieses Phänomen, welches Phillips (2014) in Bezug auf die Musik von Computerspielen bezieht, lässt sich auch auf andere Audioaspekte eines Games-Projektes übertragen. Zum Beispiel werden bei einem *Beat 'em up* Treffer erwartet, die markant, laut und durchsetzungsfähig sind (Leenders, 2012).

Es ist ausdrücklich nicht gemeint, dass alle Spiele eines Genres gleich klingen, oder zu klingen haben und der Sound Designer in seiner Kreativität beschnitten ist; vielmehr ist es als Summe verschiedener Faktoren zu verstehen⁶, die erklären, warum viele Spiele eines Genres, zum Teil eine hohe klangliche Ähnlichkeit aufweisen.

Deshalb ist es sinnvoll, als Tonschaffender mit dem eigentlichen Genre der Produktion und eventuell auch den übergeordneten, vertraut zu sein.

2.1 Vorarbeit: Übergeordnete Genres des Survival

Im Folgenden sollen nun vier Genres grob vorgestellt werden, die gewissermaßen als Elterngenres der Survival Games verstanden werden können, beziehungsweise ähnliche Merkmale wie diese aufweisen.

⁵ Ein Beispiel: „Wurde *Pac-Man* (1980) bei seinem Erscheinen noch als Actionspiel betrachtet, wird es heute als Puzzle- oder Geschicklichkeitsspiel angesehen.“ (Leenders, 2012, S. 18)

⁶ Zu diesen Faktoren zählen unter anderem die Zielgruppen, bzw. demografische Verteilung der Konsumenten in Bezug auf die Genres (Phillips, 2014).

Diese Betrachtung ist bewusst generell gehalten, da die auditive Form sehr stark vom jeweiligen Spiel und dem jeweils aktuellen Zeitgeist abhängt; zum Teil auch vom Kulturkreis in welchem es entwickelt wurde (Phillips, 2014). Vielmehr sollen die aufgezeigten Ähnlichkeiten als eine vereinfachte Momentaufnahme der Trends verstanden werden, zumal die Betrachtung als eine Basis zum Verständnis der Survival-Games dienen soll.

2.1.1 Action

Actionspiele zeichnen sich durch eine hohe Spielenergie aus. Diese kann, wie im Beispiel der *Ego-Shooter*, dadurch zustande kommen, dass viele Gegner getötet werden müssen. Rätsel, falls überhaupt vorhanden, fallen simpel aus, genauso wie der Plot; die Action steht im Vordergrund. Prominenter Vertreter, zudem mit einer offener Welt, stellen die jüngeren GRAND THEFT AUTO Spiele dar (Leenders, 2012).

Aufgabe der Vertonung ist es, diese Energie zu unterstützen. Dementsprechend sind insbesondere rhythmusorientierte und treibende Musik unterschiedlicher Stilrichtungen (Phillips, 2014) und, wie im bereits erwähnten Beispiel des *Beat 'em up*, durchsetzungsfähige Sounds, die auch im Eifer des Gefechtes zuverlässig wahrgenommen werden können, vorherrschend (Leenders, 2012).

2.1.2 Adventure

In *Adventures* stehen, im Kontrast zu Actionspielen, die Geschichte und das Erkunden, beziehungsweise das Untersuchen der Umgebung im Vordergrund. „Der Spieler muss durch das Lösen von Rätseln und durch Kombinationsgabe dieser Geschichte vorantreiben.“ (Leenders, 2012, S. 19). Bereits in einer der älteren Formen der *Adventures*, dem sogenannten *Point & Click Adventure* können und/oder müssen Gegenstände kombiniert oder gar erschaffen werden, um das Spiel voranzutreiben. Ein ähnliches Prinzip ist ein prominenter Teil von Survival-Spielen.

Da diese Art von Spielen in der Regel ein deutlich langsames Tempo einschlägt, verwundert es nicht, dass die Vertonung dezenter ausfällt. Die Musik hat die Aufgabe die Story zu unterstützen und nicht zu aufdringlich zu sein, damit das Lösen der Rätsel nicht unnötig behindert wird. In modernen *Adventures* ist desweilen ein Hang hin zu dramatischer orchestraler Musik zu erkennen (Phillips, 2014). Auch in Sachen Sound Design ist tendenziell weniger mehr. So würden die überzogenen Sounds aus dem Actionmilieu hier negativ unpassend wirken.

Bekanntere Beispiele des Genres sind: THE SECRET OF MONKEY ISLAND (1990) und LIFE IS STRANGE (2015).

2.1.3 Action-Adventure

Die Kombination der beiden Genres hat sich im Laufe der Zeit zu einem eigenen entwickelt und wird als Action-Adventure bezeichnet. „Der Übergang von Action- zu Action-Adventure-Spielen ist recht fließend. Ausschlaggebend ist ein größerer Fokus auf das Lösen von Rätseln und Geschicklichkeits[passagen][...].“ (Leenders, 2012, S. 19) Man kann also sagen, dass Action-Adventures den Forschungsgeist des Adventures mit der schnellen Reaktionsfähigkeit der Actionspiele kombinieren (Phillips, 2014).

Die Hybridform des Genres spiegelt sich auch in der Vertonung wieder. So verbinden Action-Adventures laut Phillips (2014) ruhige und treibende Musik. In Sachen Stil, seien aktuell orchestrale Richtungen, zum Teil mit Rock-Elementen, häufig der Fall. Das Sound Design hat sich an die jeweilige Spielsituation anzupassen, um immersiv⁷ zu bleiben.

Action-Adventure Titel sind zum Beispiel LUIGI'S MANSION (2002) und die bereits erwähnte TOMB RAIDER Reihe.

2.1.4 RPGs

Rollenspiele, kurz auch RPGs, haben häufig einen ähnlichen Fokus wie Adventures. Der Plot ist ein Kernelement. Als zweites kommt hier jedoch noch ein Fokus auf die Charakterentwicklung im Verlauf des Spieles hinzu. Der Spieler kann den Charakter – in manchen Spielen freier als in anderen – nach seinem Belieben personalisieren und formen. Häufig genutztes Mittel sind Level (Leenders, 2012); damit verbundenen sind auch das Erlernen von neuen *Skills* die oft in einem *Skill Tree*⁸ organisiert werden.

Da RPGs ein sehr beliebtes Genre darstellen, überrascht es nicht, dass viele Spieleentwickler Rollenspielelemente auch in andere Genres übernommen haben. Eine solche Charakterentwicklung mit neuen Fertigkeiten ist auch in so manchem Survival-Spielen wiederzufinden (z.B. THE LONG DARK (2014)).

Die Musik in RPGs kann verschiedene Funktionen übernehmen. So kann sie die Spielumgebung widerspiegeln, was besonders gut in THE WITCHER 3: WILD HUNT (2015) zu erkennen ist. Zusätzlich, oder alternativ, sollen sie die Stimmung wie in einem Film prägen. So erwarten Spieler in westlichen Fantasy-RPGs à la THE ELDER SCROLLS V: SKYRIM (2011) epische, orchestrale Musik, die an die DER HERR DER RINGE Verfilmungen erinnert (Phillips, 2014). Auch das Sound

⁷ Immersion bezeichnet den Zustand des Eintauchens in die virtuelle Welt (Phillips, 2014). Siehe auch Kapitel 3.1.1.

⁸ Ein Skilltree (engl. Fertigungsbaum) beschreibt die Hierarchie der einzelnen Skills, also wie verschiedene Skills zusammenhängen (welche Skills der Charakter bereits erlernt haben muss, um einen höherstufigen zu erlernen, etc.).

Design orientiert sich in solchen Spielen, zum Teil stark, an filmischer Ästhetik, beziehungsweise einem (film-)realistischen Ansatz.

2.2 Das Survival-Genre

Das Überleben der Spielfigur oder des Charakters als Ziel, reicht bis an die Anfänge der Computerspielgeschichte zurück. Dieses Merkmal reicht also nicht, um Spiele der Survival-Kategorie zuzuordnen; sonst wären PAC-MAN, die meisten Shooter und gewissermaßen auch jedes Strategiespiel ein Survival-Spiel.

Auch wenn es bereits zuvor Spiele mit ähnlichen Mechaniken gab⁹, gilt DAYZ (2012) – welches ursprünglich als eine Mod von ARMA II (2009) entstanden ist – als Begründer des Genres, so wie wir es heute kennen (Smith, 2014). DAYZ ist dabei ein reines Multiplayer-Spiel.

DAYZ spielt in einer Zombieapokalypse, in welcher der Spieler an einem zufälligen Ort, mit wenigen Vorräten startet. Der Twist hierbei ist, dass der Charakter neben den Zombies auch der Gefahr des Verhungerns, Verdurstens und Sterbens an verschiedenen Krankheiten ausgesetzt ist. Bedingt durch die Knappheit der Ressourcen, sind auch Begegnungen mit anderen Spielern potentiell gefährlich. Eine Story gibt es nicht direkt; vielmehr erleben die Spieler, bedingt durch die Spielumstände, sehr individuelle Geschichten (Day Z, 2017).

Aus DAYZ gehen die Kern-Mechaniken von Survival-Games hervor: Der Spieler erkundet eine Welt, in der er stets der Gefahr des Bildschirmtodes ausgesetzt ist. Es muss keine, kann aber eine personifizierte Gefahr geben, die es entweder zu vermeiden, oder zu bekämpfen gilt. Ressourcen müssen gesammelt und neue Gegenstände hergestellt werden, um das weitere Überleben zu gewährleisten (Lane, 2013). Dies wird als *Crafting* bezeichnet. Das Survival-Genre wird deshalb auch häufig als *Crafting-Survival* bezeichnet, um es vom *Horror-Survival* abzugrenzen.

Charakteristisch für beide ist eine Ressourcenknappheit, die zusätzlich dadurch verschärft werden kann, dass diese nach dem Einsammeln nicht mehr respawnen¹⁰ (Brenna, 2015). Wohingegen im Horror-Survival typischerweise nur Munition und Medipacks¹¹ als Ressourcen dargeboten werden, sind im Survival zusätzlich die Grundbedürfnisse zu befriedigen. Survival-Games grenzen sich also dahingehen vom Horror-Survival ab, dass eine Inaktivität des Spielers, auch in einer ansonsten sicheren Umgebung, zum Tod der Spielfigur führen kann (Smith, 2014). Dadurch ist

⁹ zum Beispiel MINECRAFT (2012) und UNREAL WORLD (Smith, 2014).

¹⁰ Erscheinen eingesammelte Gegenstände nach einer Zeitspanne wieder am Fundort, spricht man auch vom *Respawnen* (engl. spawning = erzeugen).

¹¹ Ein Medipack, oder auch Medikit, regeneriert die Lebenspunkte der Spielfigur. Häufig in Form eines Erste-Hilfe-Kastens, etc.

der Spieler früher oder später gezwungen, sich aktiv in potentielle Gefahr zu begeben (Lane, 2013), während er auf der anderen Seite durch das Einrichten einer sicheren Basis als Rückzugsort und Lagerstätte (die Tragekapazität ist meist begrenzt), gewissermaßen eine starke Bindung zu einem Ort aufbaut; ironisch, wenn man bedenkt, dass häufig eine Open-World und all ihre Erforschungsmöglichkeiten frei stehen (Smith, 2014).

Aus diesen Punkten ergibt sich die Notwendigkeit eines durchdachten Vorgehens und Bewegens innerhalb der Spielwelt. Hierdurch wird letztlich den Entscheidungen des Spielers eine größere Tiefe bemessen (Smith, 2014). Es besteht zudem nicht die Frage, ob der Spieler die Ressourcen finden, sondern ob er derer habhaft werden kann. Häufig wird die Schwierigkeit zusätzlich erhöht, indem Lebensmittel mit der Zeit verderben und/oder die Ausrüstung sich abnutzt. Der Spieler hat, zu keiner Zeit im Spiel, die volle Kontrolle über das Geschehen, sondern nur darüber ob und wann er aktiv wird und wann er seine gesammelten Ressourcen verwendet (Burford, 2015).

Der Reiz des Genres liegt nach Burford (2015) in der Verletzlichkeit der Spielfigur und dem Hochgefühl, welches entsteht, wenn der Spieler allen Widrigkeiten erfolgreich die Stirn bietet. Zusätzlich lässt sich anmerken, auch wenn die meisten Survival-Games eine grobe Rahmenhandlung bieten – in *THE FOREST* (2014) ist man zum Beispiel auf der Suche nach dem eigenen Sohn – sind es die, aus dem jeweiligen Spieldurchlauf entstehenden Geschichten und Begebenheiten, die das Spiel befeuern. Oft werden diese durch das Prinzip des *Permadeath*, welches im Survival weit verbreitet ist, verstärkt: Stirbt der Charakter, so ist er und der gesamte Spielfortschritt unwiederbringlich verloren; der Spieler beginnt ein neues Leben, wieder mit minimaler Versorgung.

Burford (2015) merkt an, dass sich auch das reguläre Survival-Genre (also auch nicht Horror-Survival Spiele) mit Horror überschneidet. Er sagt dazu:

The ultimate goal of any game is to create some sort of emotion within the player, whether that's the thrill of victory, sadness at a tragic narrative, enjoyment when faced with a humorous situation. Horror games are a genre dedicated to terror and thrills. But what about survival games? I believe they overlap with horror — players experience tension as they struggle to survive and thrills when they overcome a nasty obstacle, whether that's finding a scrap of food at the last second or downing a monster with your last bullet. (Burford, 2015)

An dieser Stelle sei gesagt, da in Survival-Spielen alleine das am Leben bleiben schwer ist und auf viele Charakterstatus gleichzeitig geachtet werden muss, handelt es sich um ein sehr einsteiger-unfreundliches Genre. Die Aufgabe des Sound Designers ist es also, neben dem Etablieren der Welt und der Unterstützung der Immersion, zusätzlich auch, durch passende Sounds, die Usability zu verbessern.

3 Vor der Vorproduktion: Theorie

Bevor die eigentlichen Produktionsprozesse genauer unter die Lupe genommen werden, sollen zunächst mögliche Rollen von Gameaudio diskutiert, einige Begriffe definiert und auch Erkenntnisse aus anderen Bereichen untersucht werden, um ein theoretisches Fundament zu schaffen, welches die spätere Produktion erleichtert.

3.1 Mögliche Rollen von Game-Audio

Die Möglichkeiten der Computerspielvertonung und deren jeweilige Rolle, sind in etwa so vielfältig wie die Genres. So wie Spielmechaniken kombiniert werden können, können auch die Aufgaben des Tons vielfältig miteinander vermischt werden. Dennoch kann man einige grundlegende, wiederkehrende Funktionen benennen. Ganz generell lässt sich sagen, dass folgende drei Informationen immer dem Spieler via Audio mitgeteilt werden sollen (Cullen, 2016):

- In welchem Umfeld befindet sich der Spieler? Die Welt soll abgebildet und der Spieler in die Immersion entführt werden. Dazu gehören zum Beispiel auch Wetter, Hinweise auf die Kultur oder Landschaft, etc.
- Der Zustand des Charakters. Wurde zum Beispiel ein Level-Up erreicht, oder wie viele Lebenspunkte verbleiben dem Charakter?
- Die Stimmung sowohl der aktuellen Spielsituation, als auch des gesamten Spiels. Befindet sich der Spieler etwa in einer ruhigen Umgebung, oder ist er von Feinden umgeben; befindet er sich in Gefahr oder in Sicherheit?

Sounds stellen den einfachsten Weg in die Gefühlswelt des Spielers dar, da sie beim Hörer immer auch Emotionen, beziehungsweise „innere Bilder“ (Weiss, 2015, S. 25) entstehen lassen. Visuelle und auditive Wahrnehmung stehen indes im Gegensatz zueinander, da der Mensch keine Fähigkeit besitzt, sich vor einem Klangobjekt zu verschließen; er kann zwar wegsehen, aktiv weghören jedoch nicht.

Es ist wichtig, sich als Tonschaffender stets darüber im Klaren zu sein, dass eine Vertonung immer wirkt, „egal ob man den ‚richtigen‘ oder ‚falschen‘ Sound kreiert oder abspielt. Wenn es der ‚falsche‘ Sound ist, entfaltet er nicht einfach keine Wirkung, sondern die falsche. Und das in aller Intensität.“ (Weiss, 2015, S. 35)

An dieser Stelle soll nun tiefer gegangen werden, um ein besseres Verständnis für die Wichtigkeit und auch die Funktionsweise von Game-Audio zu erreichen. Als erstes soll dazu die Immersion genauer beleuchtet werden.

3.1.1 Immersion

3.1.1.1 Glaubwürdigkeit versus Realismus

Computerspiele unterscheiden sich dahingehend von anderen Medien, dass die Immersion keine passive, sondern eine aktive Erfahrung ist. Zur Verdeutlichung zwei Beispiel: Der Zuschauer eines Films, der vergisst, dass er sich in einem Film befindet und auf einer mentalen Ebene in das Filmgeschehen involviert ist; oder aber der Gamer, der das Bewusstsein über die Steuerung, die Spielmechaniken, etc. überwindet und völlig in die Spielwelt eintaucht. Dies nennt man Immersion. In Games wird also die fiktive Welt und/oder Handlung glaubhaft; der Spieler ist umgeben von ihr und wird, metaphorisch gesprochen, eins mit ihr. Dies kann so weit reichen, dass er seinen Spielcharakter als Repräsentation seiner selbst wahrnimmt. Das bedeutet aber auch: schafft es ein Spiel nicht, den Spieler von der Echtheit der Spielerfahrung zu überzeugen, wird er nicht eintauchen können. Ist ein immersives Spielerlebnis das Designziel, so gilt es diese Immersionshemmnisse abzubauen (Phillips, 2014).

Das heißt nicht, dass eine Spielerfahrung die Realität abbilden muss, um immersiv zu sein. Ansonsten würde dies bedeuten, dass zum Beispiel Science-Fiction- oder Fantasy-Spiele nicht immersiv sein könnten, was nicht den Tatsachen entspricht. Abgesehen davon streben viele Spiele keinen direkten Bezug zur echten Welt an, man denke nur an PAC-MAN. Doch selbst im Fall, dass ein direkter Bezug zur realen Welt hergestellt wird, ist es wichtig zu bedenken, dass die Möglichkeiten der Realitätsabbildung stark begrenzt sind, bedingt durch beispielsweise dem Interface, oder aber der vereinfachten Logik von Abläufen innerhalb der Spielwelt. „Durch diese Faktoren muss sich der Spieler auch bei Spielen, die sich auf die reale Welt beziehen, immer mit einer Pseudorealität auseinandersetzen, deren Gesetzmäßigkeiten nur zum Spielzeitpunkt gelten.“ (Leenders, 2012, S. 36 f.)

Es ist also wichtiger, Konzepte zu schaffen, die in dieser Pseudorealität als logisch begründbar und funktional gesehen werden, anstatt die Realität eins zu eins umzusetzen. Leenders (2012) nennt als Beispiel das weit verbreitete Medikit und die Heilungssounds¹², deren Klang schon beinahe einer computerspielinternen Konvention zu folgen scheint.

¹² Die Palette der Heilungssounds greift regelmäßig auf hochtönige, sanfte Klänge zurück. Dies kann an Engelschöre, Atemgeräusche, *Sparkling* oder andere Synthesizer Sounds erinnern (Leenders, 2012).

Um ein realistisches Spielgefühl in einer fiktiven Welt zu schaffen, werden Sounds auf ähnliche Art und Weise an Schallquellen gebunden, wie es in der echten Welt der Fall wäre und zusätzlich indem der Sound so konzipiert ist, dass er der Atmosphäre und der dramaturgischen Entwicklung der jeweiligen Spielwelt zuträgt (Jørgensen, 2006).

3.1.1.2 Immersion durch Game-Audio

Vachon (2009) verweist darauf, dass es – während Grafik und KI mit dem *Uncanny Valley* ihren Nemesis gefunden haben, den es zu überwinden gilt, um wirklich real wirkende Abbilder der echten Welt zu schaffen – im Bereich des Tons bereits seit längerem möglich ist, Aufnahmen zu erstellen, die als realistisch wahrgenommen werden. Hierbei ist zusätzlich jedoch anzumerken, dass der Einfluss von Hollywood, das Verständnis von *Realismus* im Tonbereich stark verzerrt hat. Als Beispiel können Pistolenschüsse gesehen werden, denen im Film durch die Tonebene mehr Kraft, Autorität und Gefährlichkeit verliehen werden soll. Die überzogene Darstellung eines Schusses, man nennt dies in der Branche auch *larger than life*, ist so gesehen eine Art Statement (Leenders, 2012).

In den Filmwissenschaften ist bekannt, dass Tönelemente ohne „realen Ursprung [...]einige Regeln und Erwartungen berücksichtigen [müssen], um glaubwürdig zu sein.“ (Flückiger, 2002, S. 277) Diese Erkenntnis kann nun auf Computerspiele angewandt werden: Solange Sounds in Computerspielen die vorher genannte *Pseudorealität* bedienen und schlüssig in das Gesamtklangbild integriert werden, ist es auch möglich „selbst vollkommen unrealistische Sounds als glaubwürdig zu etablieren und damit zur Immersion beizutragen.“ (Leenders, 2012, S. 38)

Es sollte stets bedacht werden, dass Konsumenten stets hohe Ansprüche an die Qualität der Tonarbeit stellen. Dies schließt sowohl die Audioqualität der Dateien selbst ein, wie auch Stimmigkeit und Vielfältigkeit. Zusätzlich fällt jeder Ton, der den Spieler aus der Immersion entführt, negativ auf; er kann potentiell die vom Entwickler vorgesehene User-Experience gefährden (Vachon, 2009).

Die Immersion kann also im Umkehrschluss durch ein gutes und passendes Sound Design, als auch durch entsprechende Musik gefestigt werden. Phillips beschreibt 2012 unter anderem die Konzepte “Music as a State of Mind” (S. 98) und “Music as World builder” (Phillips, 2014, S. 102):

a) Music as a State of Mind

Nach Phillips ist für das erfolgreiche Spielen mancher Spiele eine bestimmte Geisteshaltung – also ein *State of Mind* – vonnöten. Mithilfe von Musik ist es demnach möglich, den Spieler beim Spielstart abzuholen und in eben diesen Zustand zu bringen. Survival-Horror-Spiele beispielsweise, sollen davon besonders profitieren: Ziel der Musik ist es

hier nicht, den Spieler durchgängig zu erschrecken. Der Spieler soll vielmehr in eine starke Anspannung versetzt werden, sodass die einzelnen Schockmomente verstärkt wirken können (Phillips, 2014).

Bedingt durch die Ähnlichkeit der Genres, kann die Musik in regulären Survival-Spielen ebenfalls ähnliche Funktion übernehmen. Auch hier muss der Spieler in die innere Alarmbereitschaft und Anspannung versetzt werden, die, zum Beispiel in *THE FOREST* (2014) durch das Gestrandet sein auf einer, von Kannibalen und Mutanten bevölkerten Insel, ganz ohne Vorräte, benötigt wird.

Die Absicht in *THE DARK SIDE OF THE MOON* war es, ein Gefühl der weiten Leere, beziehungsweise der Einsamkeit und Beklemmnis durch die Musik zu vermitteln.

b) Music as World builder

Wenn Musik die Rolle des Weltenbauers übernimmt, so sollen sie den Realismus der Spielwelt unterstreichen und erweitern, indem sie zum Beispiel den Kulturen der Welt Tiefe verleiht. Dabei ist auch klar, dass insbesondere Spiele von diesem Konzept profitieren, die eine reichhaltig erkundbare Umgebung bieten, wie RPGs und Adventures etwa (Phillips, 2014). Phillips nennt als Beispiel das polnische Spiel *THE WITCHER* (2007), doch man kann sagen, dass die Fortsetzung *THE WITCHER 3: WILD HUNT* (2015) noch einmal mehr dieses Konzept verdeutlicht: Das Spiel findet in einer düsteren Mittelalter-Fantasy Umgebung statt. Für die Musik wurden neben Orchester, auch unter anderem die Folk-Band *PERCIVAL* engagiert, deren Stil als mittelalterlich-slawisch inspiriert beschrieben werden kann (CD PROJEKT RED [GameSpot], 2016).

3.1.2 Aurales Feedback

In Computerspielen geschieht häufig vieles mit verschiedenen Objekten, Personen oder Kreaturen simultan und das an verschiedenen Orten – sowohl im Blickfeld des Spielers, als auch außerhalb davon. Fehlt das aurale Feedback, ist der Spieler nicht nur potentiell enttäuscht, sondern er wird womöglich auch aus der Immersion gerissen (Leenders, 2012). Neben dem Erzeugen von glaubhaften Welten, ist der Ton, insbesondere das Sound Design, auch für eine verbesserte Usability und Steuerung innerhalb der Spielwelt verantwortlich.

Die Modelle des *Auditory Icons* und des *Earcons* bieten eine Erklärung, wie es möglich ist, durch entsprechende Sounds eine Fusion von Informationsvermittlung und gleichzeitiger Unterstützung der fiktiven Welt zu erreichen. Beide haben ihren Ursprung in den *Auditory display studies*, also der Wissenschaft über die Verwendung von Sound als Kommunikationsmittel in Computeranwendungen (Jørgensen, 2006). Die Grundlage hierfür besteht darin, dass der Mensch in der Lage

ist, Sounds schnell zu erfassen und zu unterscheiden, zumal „zur Wahrnehmung von Klängen und Geräuschen weniger aktive Aufmerksamkeit erforderlich [ist] als zur Wahrnehmung von Bildern oder Text.“ (Ingwersen, 2005, S. 335)

Jørgensen geht 2006 einen Schritt weiter und überträgt diese Erkenntnisse auf Computerspiele, mit der Modifikation, dass der Bereich der Computerspiele, im Hinblick auf den Einsatz von Sprache, eine Ausnahme darstellt; obwohl die Begriffe *Auditory Icon* und *Earcon* als nonverbale Informationsübermittlung verstanden werden, kann Sprache im Fall der Computerspiele hinzugezählt werden, wenn sie gewisse Voraussetzungen erfüllt.

Sounds müssen, um eine informative Rolle einzunehmen, das Wissen des Spielers erweitern. Ansonsten gehen sie in der Geräuschkulisse des Spiels unter oder fallen gar negativ auf (Ingwersen, 2005). Um eine optimale Lernkurve und Spielbarkeit zu erreichen, ist es notwendig, eine Redundanz von visuellen und auralen Systemen zu schaffen (Gaver, 1997).

In Survival-Spielen müssen die verschiedenen Bedürfnisse gleichzeitig im Auge behalten werden. Sie sind in der Regel in irgendeiner Form visuell abgebildet. In *THE DARK SIDE OF THE MOON* sind sie beispielsweise permanent im HUD unten rechts angezeigt (vgl. Abbildung 1).



Abbildung 1: HUD aus *THE DARK SIDE OF THE MOON* (Fleischmann, 2017)

Der Füllstand und die Farbe informieren über den aktuellen Stand des Bedürfnisses: Ist das Icon grün und voll, so ist es befriedigt. Bereits frühe Playtests, in denen nicht der gesamte Ton implementiert war, zeigten, dass die Spieler diese, sich kontinuierlich verändernden Statusanzeigen, früher oder später aus den Augen verloren und daraufhin den Bildschirmtod erlitten. Spätere Tests führten zu der Erkenntnis, dass durch zusätzliche Töne, die auf kritische Zustände der Bedürfnisse hinweisen, dem entgegengewirkt werden konnte. Diese Töne fallen unter die Kategorie *Auditory Icons*.

3.1.2.1 Auditory Icon

Ein *Auditory Icon* bezeichnet nach Jørgensen (2006) einen Sound, der auf der Ähnlichkeit oder Übereinstimmung zu einem realen, akustischen Gegenstück basiert und auch so wahrgenommen wird. Ein prominentes Beispiel aus der Computeranwendung stellt das Geräusch von Papierzerknittern dar, wenn der Papierkorb geleert wird (Ingwersen, 2005).

In Computerspielen treten Auditory Icons vor allem in Verbindung mit kommunikativen und auf die Quelle ausgerichteten, diegetischen Sounds auf, wie zum Beispiel Gewehrschüsse und rufende Feinde. (Jørgensen, 2006) Sie sind folgendermaßen unterteilbar:

a) Urgency

Sounds mit einer *Urgency*-Funktion sind häufig Alarm- oder Warngeräusche. Sie lassen sich konkretisieren in *Notification*-Geräusche, die den Spieler über eine veränderte Spielsituation informieren, auf die er jedoch nicht unbedingt direkt reagieren muss; er ist ausreichend die Informationen erhalten zu haben (Jørgensen, 2006). Ein Beispiel aus THE DARK SIDE OF THE MOON sind die Kommentare von Rob und Jen über die fallende Raumtemperatur.

Als zweites sind noch die *Warning*-Geräusche aufzuführen, welche auf direkte Gefahren und Bedrohungen, also die Notwendigkeit einer Spieleraktion hinweisen (Jørgensen, 2006). Das passende Beispiel aus THE DARK SIDE OF THE MOON ist das Zähneklappern Robs; erst kurz vor dem Kältetod ist es gut hörbar.

b) Response

Sounds der *Response*-Kategorie geben dem Spieler die Information, dass eine Aktion oder ein Befehl, also eine Eingabe des Spielers irgendeiner Form, vom System registriert wurde. Dazu muss ein solcher Sound unmittelbar nach der Spielereingabe erklingen. Ein offensichtliches Beispiel sind Fußschritte; der Spieler wird darüber informiert, dass und wie schnell er sich fortbewegt.

Auch die *Response*-Sounds lassen sich feiner unterteilen. Zum einen in *Confirmation/Rejection* (Jørgensen, 2006). Am besten lassen sich diese mit einem Beispiel erklären: In THE DARK SIDE OF THE MOON kann der Spieler eine Brechstange finden und mit dieser zuschlagen. Trifft er einen zerstörbaren Gegenstand, wie etwa eine Glasscheibe, so hört er einen Einschlag und das Geräusch von zerbrechendem Glas (*Confirm*). Schlägt der Spieler dagegen ins Leere, so ist nur ein Swoosh zu hören (*Rejection*).

Die zweite Art sind Sounds, die offensichtlicher auf einen kommunikativen Zweck hin angelegt wurden. Oft sind sie zwar diegetischen Ursprungs, sie scheinen jedoch nicht so natürlich wie die Sounds aus dem Brechstangenbeispiel (Jørgensen, 2006).

Als Beispiel aus THE DARK SIDE OF THE MOON kann man das Geräusch beim Crafting zählen. Es zeigt dem Spieler, dass der Charakter etwas bastelt, auch wenn diese Aktion nicht sichtbar ist. Der Sound ist dabei eher symbolischer Natur und spiegelt die eigentliche Handlung nicht realitätsgetreu wieder. Warum der Spieler diese Art von Geräuschen doch als glaubhaft einordnet, beschreibt Jørgensen wie folgt: „We see, that sound is suitable for the specific object, although not in this precise format.“ (Jørgensen, 2006, S. 2)

3.1.2.2 Earcon

Earcons sind Sounds, die symbolischer Natur und gewissermaßen abstrakt sind und kein Gegenstück in der echten Welt besitzen. Die Sounds müssen schnell erfassbar sein. Dazu müssen sie entweder so angelegt sein, dass ihre Bedeutung intuitiv erschließbar ist, oder sie müssen etabliert und ihre Bedeutung dem Spieler beigebracht werden (Ingwersen, 2005).

Häufig, aber nicht ausschließlich, treten sie in Verbindung mit nichtdiegetischer Musik oder als Interfacesounds auf. In Computerspielen kann die menschliche Stimme, für den Fall, dass sie zur Lokalisation einer Quelle dient (also der Inhalt des Gesprochenen außer Acht gelassen werden kann), als Earcon angesehen werden. Klassische Beispiele sind eine Musik bei Feinden in der unmittelbaren Umgebung und Quest-Sounds (Jørgensen, 2006), wie zum Beispiel in THE WITCHER 3: WILD HUNT (2015) zum Anzeigen eines Quest-Fortschritts, oder Abschlusses. In THE DARK SIDE OF THE MOON können die Sounds beim Schlösserknacken-Mini-Game als Earcon angesehen werden; der Spieler soll die aufblinkende Sequenz wiederholen. Um die Einprägsamkeit dieses Musters zu unterstützen, hat jede Position (oben, unten, links, rechts) je einen Ton.

3.1.3 Auditive Identifikation

Mit *auditiver Identifikation* ist in den Filmwissenschaften die Identifikation eines Objekts anhand von hörbaren Merkmalen gemeint. Nach Flückinger ist die „Identifikationsfähigkeit des auditiven Systems“ des Menschen sehr hoch (2002, S. 109). Dabei ist zu beobachten, dass mit steigendem Erfahrungsschatz bezüglich einer bestimmten Gattung von Gegenständen, Lebewesen, etc., auch die Feinheit der Unterscheidungsfähigkeit zunimmt¹³ (Flückiger, 2002).

Dabei setzt sich ein Klangobjekt stets aus *obligatorischen* und *fakultativen Merkmalen* zusammen. Erstere geben Auskunft über die Gattung des Objekts, letztere charakterisieren ein spezifisches Objekt durch Eigenschaften wie etwa Material, Masse, etc. (Flückiger, 2002). Dabei ist

¹³ Gemeint ist hiermit, dass beispielsweise ein Automechaniker potentiell in der Lage ist, verschiedene Automodelle anhand ihrer Motorengeräusche zu unterscheiden, da er mit diesen häufig in Kontakt ist und so einen hohen Erfahrungsschatz besitzt.

diese Art der Identifikation von Klangobjekten bereits Teil des täglichen Lebens und eine Etablierung ist deshalb nur bei fiktiven Geräuschquellen zur korrekten Erkennung notwendig. Der Mensch neigt dabei eher zur Generalisierung als im visuellen Bereich; häufig wird die Gattung erkannt und nicht ein spezifisches Exemplar (Flückiger, 2002).

Die „Tendenz, das Ganze vor seinen Teilen zu identifizieren“ (Flückiger, 2002, S. 112), kann sich der Sound Designer mit der Technik des Layering zunutze machen, welche vereinfacht gesagt verschiedene Einzelsounds zu einem Gesamtsound verbindet (→ Kapitel 4.2.5.2).

Der Klang eines Objekts gibt jedoch mehr preis, als nur welcher Gattung es angehört. Es werden immer auch die materielle Beschaffenheit, die Masse und das Volumen, beziehungsweise die Größe vermittelt (Flückiger, 2002). Damit geht eine Erwartungshaltung beim Konsument einher. Im Falle der Größe, geht diese auf die Schallausbreitung zurück; konkret steht sie im Zusammenhang mit der Wellenlänge: sieht der Konsument ein großes oder dickes Objekt, werden beispielsweise eher tiefere, bei einem kleinen oder dünnen Objekt eher hochfrequente Geräusche erwartet (Flückiger, 2002; Cancellaro, 2006). Ingwersen nennt 2005 noch weitere Informationen, die aus dem Klang hervorgehen:

- der Zustand eines Objektes; ist beispielsweise ein heruntergefallenes Glas am Boden zer-schellt?
- „unsichtbare Strukturen“ (S. 338); ob eine Objekt hohl ist, kann etwa durch einen Klopf-test und dem daraus resultierenden Geräusch ermittelt werden.
- „dynamische Prozesse“ (S. 338); so verändert sich beim Befüllen eines Glases die Ton-höhe mit fortschreitendem Füllstand.
- „anormale Strukturen“ (S. 338); ist zum Beispiel eine Maschine in gutem Zustand, oder ist sie defekt?

Der Sound Designer muss also dafür sorgen, dass ein Objekt im Spiel aus den richtigen auditiven Informationen zusammengesetzt ist und richtig identifiziert wird. Außerdem kann es genutzt werden, um wichtige Objekte durch fakultative Merkmale von unwichtigen zu unterscheiden. So kann sich etwa eine wichtige Tür von den anderen Türen klanglich abheben, um den Spieler in die richtige, beziehungsweise geplante Richtung zu lenken. Ein weiteres Beispiel, welches auch der Auditory Icon Gattung Response angehört, ist das Ausrüsten von Gegenständen, wie es beispielsweise in THE ELDER SCROLLS V: SKYRIM (2011) umgesetzt wurde. Ob der Spieler einen schweren Streithammer, einen kleinen und leichten Dolch oder einen hölzernen Stab in die Hand nimmt; alles ist anhand auditiver Merkmalen unterscheidbar.

Das Wissen über die vermittelten Informationen kann auch genutzt werden, um gezielt eine Dissonanz zwischen Visuellem und Gehörtem zu schaffen. Ein gutes Beispiel aus dem Filmbereich ist der *eine Ring* aus *DER HERR DER RINGE*; der Zuschauer erwartet einen kleinen, leichten Goldring. Fällt dieser jedoch zu Boden, so klingt er schwer und groß und deckt sich nicht mit der Alltagserfahrung. Es wird klar, dass dieser Ring eine ‚gewichtige Rolle‘ in der Geschichte spielt. Werden jedoch nicht mit einem gezielten Grund oder ausversehen falsche Informationen übermittelt, muss damit gerechnet werden, dass der Spieler den Sound als unpassend empfindet und potentiell aus der Immersion herausgerissen wird.

Ein Beispiel: zu Testzwecken wurden in *THE DARK SIDE OF THE MOON* vorübergehend die Geräusche beim Zerstören der Couchtische bis auf das Zerbrechen der Glasplatte stummgeschaltet. Wenn nun der Spieler einen Couchtisch in seine Einzelteile zerschlägt, sieht er wie Holzstücke herausbrechen, hört aber nur die springende Glasscheibe. Das Gesehene steht im Kontrast zum Gehörten, zumal Holz eine wichtige Ressource im Spiel darstellt. Der Spieler ist in diesem Moment verwirrt.

3.1.4 Räumliche Situierung

Sowohl die Ortung von Objekten oder Personen, als auch das Erkennen des aktuellen Raumes, sind Teil der *räumlichen Situierung* (Flückiger, 2002). Der Ton hilft dem Spieler also bei der Orientierung. Da die auditive Wahrnehmung, anders als die visuelle, rundum reicht, kann der Spieler durch Geräusche die Anwesenheit und ungefähre Position von Objekten und Events zu seinem Charakter, auch außerhalb des Sichtfelds, wahrnehmen (Jørgensen, 2006). Dies ist insbesondere in Spielen wichtig, die einen möglichen Feindkontakt in sich bergen, FPS etwa, oder auch Survival-Spiele mit Kampfsystem.

Dabei ist zu beachten, dass sich durch die Entfernung neben dem Pegel, auch der Klang und das Verhältnis von Direkt- und Diffusschall verändert. Ist der diffuse Anteil groß genug, so ist er als Hall wahrnehmbar (Flückiger, 2002). Dabei beeinflussen die Raumgröße und –form, Wand-, Boden und Deckenmaterial, vorhandenes Mobiliar, etc. die Rauminformation. Sie besteht aus dem Direktschall, den ersten Reflexionen, dem verzögert eintreffendem Nachhall (Dickreiter, 2014) und der Klangverfärbungen die durch unterschiedliches Absorptions- und Reflexionsverhalten entstehen (Flückiger, 2002). Die Nachhallzeit allein ist nicht ausreichend, um zuverlässig auf die Beschaffenheit eines Raums zu schließen. Ein Beispiel: Gegeben sei ein gekacheltes und gefliestes Badezimmer, als auch eine Halle mit derselben, relativ langen Nachhallzeit. Die frühen Reflexionen geben nun zusätzlich einen Hinweis auf die Raumgröße. Dies bedeutet, während das Badezimmer durch schnelle erste Reflexionen und schnell folgenden Hall beschrieben werden kann, besteht die Rauminformation des großen Raumes aus verzögerten ersten Reflexionen und

einem ebenfalls verzögerten Nachhall. Zudem kann davon ausgegangen werden, dass die spektralen Anteile der beiden Nachhalle unterschiedlich sein werden.

Die räumliche Ortung wird später in der Middleware, beziehungsweise der Audio-Engine umgesetzt. Es wird der Schall berechnet, der am *Listener* ankommt, sozusagen dem Gehörpunkt. Als Grundlage der Berechnung werden die Position des Listeners und die des jeweiligen Schallereignisses zurate gezogen. Die unterschiedlichen Middlewares und Audio-Engines bieten verschiedene Möglichkeiten der Umsetzung dieser akustischen Abbildung (Leenders, 2012).

Es ist also im Sinne des Sound Designers, bereits frühzeitig abzuklären, welche Anforderungen an diese Abbildung bestehen. Zum Beispiel: Welche Position soll der *Listener* haben, welche Raumtypen kommen vor, welche (Audio-)Engine und Middleware wird verwendet und welche Möglichkeiten der räumlichen Darstellung sind in diesen möglich.

3.1.5 Gameplay Identifikation

Sind in einem Spiel verschiedene Gameplay-Möglichkeiten mit eigener Steuerung angesiedelt, so kann durch die Vertonung angezeigt werden, welche die aktuelle ist. Ein Beispiel: im Kampfmodus haben Tasten eine andere Bedeutung als im regulären. Wird der Spieler nun angegriffen, wird ihm durch den Ton signalisiert, dass er die Kampfsteuerung zu nutzen hat. In diesem Fall bietet sich insbesondere die Musik und Feindgeräusche, wie etwa Ausrufe an. Andere Beispiele sind Menüs, Inventare (Phillips, 2014) und auch Mini-Games. Je nach Spiel können diverse Vertonungsmöglichkeiten angebracht sein; es muss also nicht unbedingt durch die Musik geschehen.

In THE DARK SIDE OF THE MOON wird das Öffnen und Schließen des Inventars durch ein Piepsen und das offene Inventar selbst durch ein Geräusch, das an einen Lüfter oder ein Laufwerk erinnert, dargestellt. Dem Spieler wird signalisiert, dass er sich nicht fortbewegen kann und die inventarspezifische Steuerung aktiv ist. Die Musik ändert sich in diesem Fall nicht. Ein weiteres Beispiel sind die Pieps-Geräusche beim Schlösserknacken; sie tauchen nur in diesem Zusammenhang auf. Dem Spieler wird signalisiert, dass er sich in einem Mini-Game befindet und dieses nun mit Maussteuerung zu spielen hat.

3.1.6 Stille

Mit dem Begriff der Stille, kann in Games Unterschiedliches gemeint sein. Zum einen kann er für absolute Stille, also *digital null*, oder aber auch für die Abwesenheit von Musik und/oder Atmo, oder etwa einzelner Sound-Gruppen stehen. Stille kann in Spielen, wie auch im Film als Stilmittel eingesetzt werden. Es muss jedoch bedacht werden, dass absolute Stille, mehr noch als im Film, für Verwirrung und der Vermutung eines Fehlers sorgen kann: Alle zuvor diskutierten Arten des

auditiven Feedbacks greifen nicht mehr; der Spieler bekommt in diesem Moment keinerlei auditive Rückmeldung darüber, ob seine Eingaben richtig im System angekommen sind, in welchem Gameplay-Modus er sich befindet, ob seine Aktionen erfolgreich waren und ob sich etwas außerhalb des Sichtbereichs, aber dennoch in der unmittelbaren Nähe zum Charakter befindet.

Jørgensen (2006) konnte durch Untersuchungen feststellen, dass die Immersion und damit auch das Konstrukt der virtuellen Welt beim Fehlen von Sound verbleicht und ein Spiel auf seine Spielregeln und Spielmechaniken reduziert wahrgenommen wird. Es ist deshalb sinnvoller, ein gewisses Fundament an Grundsounds zu behalten (beispielsweise Charaktersounds wie Fußschritte) und dafür andere Geräusche auszublenden, sodass ein stille-ähnlicher Zustand erreicht wird.

So kann Stille potentiell dieselben Funktionen wie im Film übernehmen. Nach Flückinger (2002) sind das unter anderem:

- a) „Stille als Todessymbol“ (S. 234) In Computerspielen wird des Öfteren nach dem Bildschirmtod auf Stille abgeblendet.
- b) „Unangenehme Gefühle, unterdrückte Konflikte, Peinlichkeit“ (S. 235).
- c) „Zirkuseffekt“ (S. 236) Die Stille wird als Ruhe vor dem Sturm genutzt, um dem folgenden, lauten Geräusch einen noch lauterem Eindruck und Gewichtung zu verleihen.
- d) „In Verbindung mit Desorientierung [...]. In der Stille macht nur irgendwo ein leises Geräusch auf den Feind aufmerksam.“ (S. 237)

Zusätzlich kann Stille in Computerspielen, wenn damit die Abwesenheit von Musik gemeint ist, dazu dienen, einer Abnutzung dieser entgegen zu wirken, oder aber auch die Funktion einer Verschnaufpause einnehmen. Ob dies passend ist, ist jedoch unter anderem vom Genre abhängig (Leenders, 2012).

3.2 Dynamische Vertonung

Auch wenn der Name es vermuten lassen könnte, meint eine *dynamische Vertonung* nicht eine Vertonung mit einem hohen Dynamikumfang und bezieht sich auch nicht auf Dynamikanweisungen in der Partitur der Spielmusik. Stattdessen ist damit eine Vertonung gemeint, die sich anhand diverser Parameter verändern kann und sich dadurch von einer *statischen Vertonung* unterscheidet. Während die statische Vertonung (wie zum Beispiel die eines Filmes) sich dadurch auszeichnet, immer gleich zu bleiben, wird durch eine dynamische Vertonung gewährleistet, dass das „

[...] Tonkonzept bei jedem Spieldurchgang neu und einzigartig ist und jedem Spieler eine ganz persönliche eigene Erfahrung bietet.“ (Leenders, 2012, S. 56 f.)

Der Bereich der dynamischen Vertonung umfasst sowohl *adaptive*, als auch *interaktive* Vertonung.

- a) *Interaktiv* ist eine Vertonung dann, wenn sie auf die Eingabe des Spielers reagiert. Schrittgeräusche sind demnach interaktiv.

- b) Eine *adaptive* Vertonung reagiert stattdessen auf Gameplay und sich verändernde Parameter der Spielwelt, statt auf Spielereingaben. Verändert sich die Schrittgeräusche durch die Bodenbeschaffenheit, oder ändert sich die Musik bei Feinden in der Nähe, so ist die Vertonung adaptiv (Horowitz & Looney, 2014).

Es lässt sich also sagen, dass ein Computerspiel, durch seine Nichtlinearität, das Einbeziehen von Parametern und Variablen und die Spielereingabe von Natur aus dynamisch vertont ist (Leenders, 2012). Dabei wird diese Vertonung in *Echtzeit* anhand dieser Einflüsse zusammengesetzt (Weiss, 2015).

3.3 Diegese in Bezug auf Games

Unter Diegese wird in der Filmtheorie die Welt und die dazugehörige Zeit verstanden, in welcher sich die Figuren des Films befinden und die Handlung stattfindet. Sound ist also dann diegetisch, wenn er Bestandteil dieser Welt ist (Flückiger, 2002). Dazu gehören Foley und beispielsweise Dialoge. Nichtdiegetisch ist nach dieser Einteilung die Filmmusik, insofern sie nicht für die Figuren hörbar ist und somit nicht innerhalb der Szene spielt. Sie beeinflusst die Handlung nicht, sondern untermalen sie nur; manchmal deutet sie den Ausgang einer Szene an. In Bezug auf Computerspiele ist dies beispielsweise anders, da nichtdiegetische Geräusche den Spieler mit Informationen versorgen. Dadurch haben sie die Macht, seine Entscheidungen und folglich die Handlung im Spiel und das Ergebnis einer Szene direkt zu beeinflussen (Jørgensen, 2006).

Dabei ist anzumerken, dass diegetisch nicht gleichzusetzen ist mit sichtbar; auch eine nicht sichtbare Schallquelle kann diegetisch sein, solange sie Teil der Szene, Umgebung, etc. ist (Berndt, 2010). Ein Beispiel aus *THE FOREST* (2014): Der Spieler kann in der Spielwelt einen Walkman und diverse Musikkassetten finden. Durch das Anhören der Musik kann die Ausdauer regeneriert werden. Um die Musik zu starten, muss der Spieler den Walkman in die Hand nehmen; er ist sichtbar. Nun steht es dem Spieler jedoch frei, ihn wieder wegzustecken ohne die Musik zu beenden; der Walkman ist nicht sichtbar, die Musik spielt jedoch trotzdem und kann auch Gegner in Hörweite anlocken, ist also Teil der Spielwelt und damit diegetisch.

Der Diegese-Begriff im Bereich des Computerspieltons kann nach Leenders (2012) durch den Einfluss des Spielers auf dessen Gestalt wie folgt erweitert und einteilen werden:

- a) *Undynamisch-nichtdiegetisch* ist derjenige Ton, der auf keine Weise vom Spieler beeinflusst werden kann und nicht Teil der Spielwelt ist, wie zum Beispiel die Musik einer Cutscene.
- b) *Adaptiv-nichtdiegetischer* Sound beschreibt solchen, der sich anhand der Veränderung von Spielparametern ändert dennoch nicht innerhalb des Spieluniversums spielt. Dies kann beispielsweise adaptive Hintergrundmusik sein.
- c) *Interaktiv-Nichtdiegetisch* sind Sounds, die zwar vom Spieler getriggert werden, jedoch nicht in der eigentlichen Spielwelt, sondern einer Zwischenebene stattfinden, wie zum Beispiel HUD- oder Menü-Sounds.
- d) *Undynamisch-diegetisch* bedeutet, dass der Sound zwar Teil der Spielwelt, jedoch nicht vom Spieler beeinflussbar ist, wie etwa eine statische Atmo.
- e) *Adaptiv-diegetische* Sounds dagegen sind Teil der Spielwelt, verändern sich aber durch Änderungen dieser. Eine Atmo die sich mit dem Tag-Nacht-Zyklus ändert wäre demnach adaptiv-diegetisch.
- f) Ein Sound ist *interaktiv-diegetisch*, wenn er eine Spielereingabe als Teil der Welt repräsentiert, wie zum Beispiel ein Sprung- und Laufgeräusch.

Es ist in Computerspielen auch nicht unüblich und durchaus von Seiten der Spieler akzeptiert, dass sich diegetische Sounds direkt an den Spieler wenden, ohne dabei seltsam zu wirken, oder wie im Film, einen höheren künstlerischen Anspruch zu haben. Viel mehr ist dies eine Möglichkeit die Usability zu verbessern und gleichzeitig die fiktive Welt zu stärken. Vor allem in Strategiespielen ist dies zu beobachten: Hier sprechen ausgewählte Truppen häufig den Spieler direkt an, wodurch der Spieler über die Truppenart und deren ausgeführten Handlung in Kenntnis gesetzt wird (Jørgensen, 2006).

3.4 Rolle von Stereotypen in der Vertonung

Im Bereich des Sound Designs werden Sounds, die werkübergreifend eine häufige Verwendung finden, als *Stereotypen* bezeichnet. Da die Stereotypen und ihre Bedeutung bereits durch ihr häufige

figes Auftreten in verschiedenen Werken etabliert sind, können sie, neben dem Einsatz von Sprache und passender Musik, helfen, dem Zuschauer zügig eine glaubhafte Welt zu vermitteln. An Stereotypen sind immer auch Assoziationen gebunden (Weiss, 2015), da Stereotypen in der Regel im Kontext eines Genres oder einer Narrative stehen. Durch ihr häufiges Auftreten, geraten sie in den Hintergrund und werden oft nur unterschwellig wahrgenommen (Flückiger, 2002). Das aus der Filntontheorie stammende Konzept der Stereotypen lässt sich auch auf andere audiovisuelle Medien anwenden; auch bei Computerspielen lassen sich stereotypische Sounds finden. Wenn man sich beispielsweise die Schlaggeräusche eines der Beat 'em up Spiele der TEKKEN-Reihe anhört, ist klar, dass Action- oder Martial-Arts-Filme als Inspiration des Sound Designs gedient haben.

Stereotypen haben durch ihre verallgemeinernde Natur die Macht, auch kompliziertere Zusammenhänge oder Begebenheiten zu vereinfachen. Sie können dadurch die Orientierung innerhalb einer Szene oder Situation verbessern. Dies kann zum Beispiel durch eine Überbetonung von Fraktions- oder Gruppenzugehörigkeiten geschehen: Die Feindgebiete können mit besonders düsteren, angsteinflößenden und basslastigen Atmos vertont werden, während die der Verbündeten insgesamt durch positive Tonuntermalung gekennzeichnet sein können und dadurch Sicherheit ausstrahlen (Flückiger, 2002). In diesem Zusammenhang kann man auch feststellen, dass Sounds allgemein (also nicht nur Stereotypen) eine ähnliche Funktion wie die eines *Leitmotivs*¹⁴ übernehmen können (Flückiger, 2002). Ein Beispiel hierfür sind quietschende Türen im Thriller- oder Horror-Genre. Sie sind als Stereotyp Teil des Genres und vermitteln die Zugehörigkeit des Werkes zu eben jenen. Dem Zuschauer oder Spieler wird unterschwellig vermittelt, welcher Natur das vorliegende Werk ist. Zudem übernimmt die quietschende Tür eine leitmotivähnliche Funktion, indem sie die Gefahr, die mit der quietschenden Tür einhergeht, in die Bedeutung des Geräusches mitaufnimmt und sich so von allen anderen Türen des Werkes unterscheidet. Es lässt sich also sagen: „[...] das Klangobjekt allein kann den symbolischen Komplex repräsentieren – und die Bildung einer übergeordneten dramatischen Struktur.“ (Flückiger, 2002, S. 187)

Durch den Einsatz von Stereotypen wird dem Spieler, dank deren Vertrautheit, ein gewisses Maß an Sicherheit vermittelt, da sich die Sounds in einem strukturellen Rahmen bewegen. Nach Flückiger (2002) profitieren insbesondere extremere oder brutalere Genres und Werke vom regen Stereotypeneinsatz, wie etwa Horror und Science-Fiction.

¹⁴ Ein Leitmotiv ist ein Motiv eines Musikstücks, welches eine Person, einen Ort, eine Situation, etc. musikalisch charakterisiert. Seine Variationen innerhalb des Stücks stellen die Wandlung des jeweiligen Subjekts dar. (Neubauer, 1994, nach Flückiger, 2002, S.183)

Da Survival-Games, wie bereits diskutiert, artverwandt mit dem Horror-Bereich sind, kann es sinnvoll sein, auf Sound-Stereotypen zurückzugreifen. Zudem kann die, durch die Spielmechaniken bedingte, hohe Komplexität verringert werden. Dadurch kann einer Überforderung des Spielers entgegengewirkt und potentiell spielleistungsverschlechternder Stress, sowie eventuell entstehende Frustration verringert werden. Ein übermäßig frustrierter Spieler ist immer auch potentiell einer, der das Spiel abbricht und nicht wieder aufnimmt.

THE DARK SIDE OF THE MOON ist ein Survival-Spiel in einer Science-Fiction-Umgebung. Im Bereich der Science-Fiction hat besonders STAR WARS die Türgeräusche geprägt. So folgen seither die hydraulischen Schiebetüren von Raumschiffen, mal mehr und mal weniger offensichtlich, diesem akustischen Kodex. Die Türen im Spiel sind ebenfalls von diesem Türgeräusch-Urvater inspiriert. Dem Spieler wird beim Durchschreiten der Türen durch das Geräusch immer wieder subtil verdeutlicht, dass er sich auf einem Raumschiff in einer fernen Zukunft befindet und nicht durch die Schiebetür einer U-Bahn tritt. Die Musik und Atmo wird beim Verlassen des sicheren Raumes zudem bedrohlicher und signalisiert dem Spieler, dass er sich potentiell in Gefahr befindet. Erst wenn er sich wieder in einem der Hotelzimmer befindet, in denen er ein Lager aufschlagen kann, nimmt diese Bedrohlichkeit wieder ab. Ziel der Tongestaltung war es also, dass der Spieler die Bedeutung der Sounds nicht bewusst wahrnimmt, sondern sie ihn auf eine emotionale Weise unterbewusst unterstützen.

4 Die Produktion

Die Entwicklung eines Spieles besteht aus Sicht eines Sound Designers aus den Phasen Vorproduktion, Sound Design (im Sinne der eigentlichen Schaffung der Sounds), sowie Implementierung innerhalb der Middleware. Voice Over und Musikproduktion können je nach Produktion hinzukommen. Die Einbindung in die Game-Engine gehört hingegen zum Aufgabenbereich des Audioprogrammierers. Es ist wichtig für die Vertonung, dass der Tonschaffende so früh wie möglich in den Produktionsprozess eingebunden wird. Da möglicherweise notwendige Änderungen erst beim Testen innerhalb des Spieles sichtbar/hörbar werden, also erst im weit fortgeschrittenen Stadium einer Produktion, sind diese aus zeitlichen Gründen nur durchführbar, wenn der eigentliche Kern des Sound Designs und der Implementierung bereits gestemmt wurde. Können diese wichtigen Änderungen nicht mehr vorgenommen werden, hat darunter schlussendlich das gesamte Spielerlebnis zu leiden.

4.1 Vorproduktion

Egal in welcher Konstellation man zur jeweiligen Produktion hinstößt, sei es als fester Bestandteil des Teams von Anfang an, oder als freier Mitarbeiter mitten in der laufenden Produktion, ist es essentiell, vor dem eigentlichen Sound Design, einige wichtige Dinge abzuklären. Diese Schritte werden als Vorproduktion bezeichnet.

An erster und oberster Stelle steht die Kommunikation mit den verschiedenen Departments.

4.1.1 Kommunikation mit dem Team: Producer und Game-Designer

Eine der ersten Anlaufstellen sollten die Verantwortlichen und/oder die Game-Designer sein, da von ihnen einige grundlegende Informationen benötigt werden.

Die offensichtlichste Frage ist: Was für ein Spiel ist geplant? Das Ziel dieser Frage ist es, zum einen das Genre und die Thematik des Spieles zu erfahren und als weiterführenden Schritt ein genaueres Verständnis der Spielmechaniken zu erhalten (Marks, 2017). Diese Informationen sind für gewöhnlich in einem *GDD*, einem *Game-Design-Document* festgehalten. Es gibt keine Standardisierung für GDDs, weshalb sie in jeder Produktion unterschiedlich ausfallen können. Dieses GDD ist für alle Departments überaus wichtig; in ihm sind idealerweise sämtliche Regeln, Spielmechaniken und Attribute aufgeführt und erläutert. Ein GDD wird über die gesamte Produktionsphase ständig weiterentwickelt und verändert, weshalb es wichtig ist, regelmäßig den neuesten Stand zu überprüfen (Phillips, 2014).

Ähnlich hilfreich um einen Einblick in die Stimmung des Spiels zu erhalten, ist ein Dialog- oder Voice-Over-Skript, falls vorhanden. Durch den Blick auf das gesprochene Wort werden Plot und Charaktere greifbarer.

Falls es bereits eine *Asset-Liste* für den Ton gibt, ist sie in der Regel von diesem Department zu erhalten; sie beinhaltet alle für das Spiel benötigten Sounds (Horowitz & Looney, 2014). Falls es solch eine Liste nicht gibt, kann es durchaus die Aufgabe des Tonschaffenden sein, eine solche zu erstellen. Selbst wenn es nicht explizit als Aufgabe zugeteilt wurde, empfiehlt es sich, die erforderliche Zeit in die Anfertigung einer solchen zu investieren, da sie für den Sound Design Prozess als Wegweiser dient und so die Gefahr Sounds zu vergessen minimiert wird.

Weitere wichtige und sinnvolle Informationen, die im Gespräch erhalten werden können sind (Marks, 2017):

- Welche anderen Spiele, Filme, etc. dienen den Producern und Game-Designern als Inspiration, oder haben als solche gedient?
- Ist bereits ein bestimmtes Musik-Genre geplant?
- Für welche Plattform ist das Spiel geplant? PC, Konsole (wenn ja, welche), Mobile, etc.?

Der letzte Punkt hängt eng mit den technischen Rahmenbedingungen und Begrenzungen zusammen, auf welche im Kapitel 4.1.4 näher eingegangen wird.

Ein Auszug aus dem GDD von THE DARK SIDE OF THE MOON, welcher drei der Kernmechaniken – Sättigung, Hydratation und Körpertemperatur – betrifft, soll an dieser Stelle als Beispiel folgen (Fleischmann & Wagner, 2017, S. 3 f.)¹⁵:

b) Satiation (how much hunger remaining)

A calorie-counter is running in the background, displayed to the player as i.e. 78% satiation. The character needs 3000 kcal per day. His need may vary, depending on the kind of work he is doing. Running consumes more calories than walking, walking more than standing or sleeping. Satiation increases when consuming food that contains calories (anything except for water. The percentage depends on the food. When having 95% satiation and consuming something that replenishes 10%, the item will remain in the inventory with having 5% left.

¹⁵ Bei dem GDD handelt es sich um ein internes Dokument, weshalb das vollständige Dokument nur für das Produktionsteam zugänglich ist.

- 30% remaining: Malus sets in, increases the closer the satiation gets to 0% (Loss in energy, Damage over Time, the initial damage is 0,5% of max HP per second, malus increases linear with 5% per -10% satiation)
- 0% starvation

c) Hydration (grade of thirst)

The character needs 3 liters of water (or other drinks) per day. The amount is also represented by a percentage on the thirst bar.

- 50% remaining: Rob comments on being thirsty
- 40% remaining: Malus sets in, increases the closer the hydration gets to 0% (Loss in energy, Damage over Time, initial damage = 1% max HP/sec, malus increases x^2 per -10% hydration) ((blurry vision))
- 0% death of thirst

d) Body Temperature

Rob's body temperature is displayed in °C. Rob's body temperature is ideally 37°C.

Regenerates if character is in a warm area (slowly in warm areas, faster when a fire is lit). Temperature decreases when not being in a safe room with an ongoing fire during the night phase.

BODY TEMPERATURE (°C)	MALUS
37 - 36	None
35 – 32	Rob shivers, Movement Speed is decreased, frost flakes appear around screen
33 – 30	Damage over time: 0,1% max HP / sec (constant damage)
30 – 28	camera sets in and out of focus, frost flakes expand on the screen, tunnel vision & dark vignette, 5% max HP / sec Damage over Time (constant damage)
26	instant death

Aus diesem Auszug wird einiges im Hinblick auf die benötigten Audio-Assets ersichtlich. Der Spieler kann beispielsweise sowohl Nahrung als auch Getränke aufsammeln und zu sich nehmen. Er kann sowohl gehen als auch rennen, außerdem ist er in der Lage ein Feuer zu entzünden. Da es sich um ein Survival-Spiel handelt, ist es sehr wahrscheinlich, dass für das Feuer eigene Ressourcen gesammelt und kombiniert werden müssen, um dieses zu craften. Es gibt kritische Werte für die Attribute, auf die zusätzlich durch ein Audio-Feedback in Form eines Spruches und im Fall der Körpertemperatur zusätzlich durch das Geräusch klappernder Zähne hingewiesen werden soll. Zudem gibt es Tag- als auch Nachtphasen.

4.1.2 Kommunikation mit dem Team: Art-Department

Bilder sagen mehr als tausend Worte. Getreu diesem Motto ist es Ziel der Kommunikation mit dem Art-Department, eine genauere Vorstellung der Visualisierung des Spiels zu erhalten und daraus Inspiration zu schöpfen.

Die *Art-Bible* ist hilfreich, um der allgemeinen Atmosphäre näher zu kommen. Hier listet der Art-Director die Richtlinien auf, nach welchen der Look des Spiels umzusetzen ist, um so ein einheitliches Erscheinungsbild zu gewährleisten (Chadwick, 2015). Hier kann zum Beispiel beschrieben sein, wer die Charaktere sind und wie sie aussehen, welche architektonischen Stile auftreten, welche Materialien vorkommen und welche davon gewöhnlich und welche wertvoll sind.

Wertvoll für den Sound Designer sind auch *Mood Boards* und die *Concept-Art*, welche Bestandteil der Art-Bible sein können. Ersteres bezeichnet eine Bild-Collage, welche zur Inspiration und zum Einfangen der Spielatmosphäre oder Emotion dient (Lennartz, 2009), letzteres die ersten Konzeptzeichnungen; Mood Boards können neben Bildern anderer Produktionen auch bereits Concept-Art beinhalten. Anhand der Concept-Art können zum Beispiel Mechanismen von Gegenständen und Maschinen des Spiels nachvollzogen werden und auch ein Eindruck der Charaktere und deren Bekleidung gewonnen werden. Durch die bildliche Darstellung der Charaktere lassen sich Einblicke in die innere Motivation und deren Umgang gewinnen. Anhand dieser Informationen kann sich der Sound Designer also ein gutes Bild davon machen, wie das Spiel später klingen soll (Phillips, 2014).

4.1.3 Kommunikation mit dem Team: Programmierung

Für den Sound Designer stellt der Audio-Programmierer einen wichtigen Partner in der Vertonung eines Computerspiels dar. Es ist möglich, dass er der Ansprechpartner in Belangen zu technischen Grenzen oder Bedürfnissen der Produktion ist (Horowitz & Looney, 2014). Sicher ist jedoch, dass er für die Implementierung der Sounds in die Engine verantwortlich sein wird. Es

muss also geklärt werden, welche Interaktionen wie umgesetzt werden und was dies für die Vertonung bedeutet. Sollen die Fußschritte beispielsweise als *Loop* mit Abbruchbedingung, also als Dauerschleife abgerufen werden? Oder sollen die Trigger an die Animation angebunden werden und die Sounds jeweils als einmalig abspielender *One Shot* vorliegen?

Das Ziel ist es, eine Art Datenstruktur zu erschaffen, die für beide Seiten praktikabel ist. Wie später noch genauer erläutert wird, kann die Strukturierung der Audio-Banks auch Auswirkungen auf die Performance des Spieles haben (Sound Librarian, kein Datum) (→ Kapitel 5.1.12).

Durch die Aufgabenteilung zwischen Sound Designer und Programmierer, kann einiges an Zeit gespart werden; in kleineren Teams kann es durchaus vorkommen dass die Audio-Programmierung keinem spezialisierten Audio-Programmierer zugewiesen wird. In diesem Fall kann der jeweilige Programmierer nach dieser ersten Planung am Source-Code mitarbeiten, bis die Implementierung in der Middleware durch den Sound Designer umgesetzt ist. Er kennt sich also bereits im Code aus und weiß an welcher Stelle die jeweiligen Methoden zu setzen sind.

Um Fehler und Frustration zu vermeiden, müssen beide Seiten stets auf demselben Wissensstand sein. Es ist insgesamt wichtig, dass der Tonschaffende stets über Planänderungen, die in irgendeiner Weise die Vertonung verändern könnten, informiert wird – insbesondere bezüglich der Umsetzung der Mechaniken und der Interaktion. Genauso ist es die Pflicht des Sound Designers, die Programmierer über Planänderungen zu updaten. Der Sound Designer kann zum Beispiel innerhalb einer Middleware Parameter benennen, setzen und diese selbstverständlich auch ändern. Zwei Beispielszenarien, die so in *THE DARK SIDE OF THE MOON* hätten auftreten können:

- a) Die Fußschritte werden einzeln getriggert. Der Trigger wird nach einem definierten Zeitintervall von der Unreal-Engine 4 gefeuert. Nun sei die Situation diejenige: Die Programmierung entscheidet, etwa aus Gründen der Übersichtlichkeit im Code, das Gehen als Schleife zu gestalten (der Charakter bewegt sich zum Beispiel nach vorne, solange die W-Taste gedrückt ist), ohne den Trigger zu verändern. Wird die Umsetzung in der Middleware nicht angepasst, so ist nur ein erster Fußschritt zu hören. Der Sound Designer muss einen Loop mit Parameter erstellen, dessen Werte der Programmierung bekannt sind (z.B. `IsWalking` 0 = Stehen, 1 = Laufen).

- b) Sprunggeräusche sind in zwei separaten Events umgesetzt: eins für den Absprung `JumpUp` und das zweite für die Landung `JumpLand` (vgl. Abbildung 2).

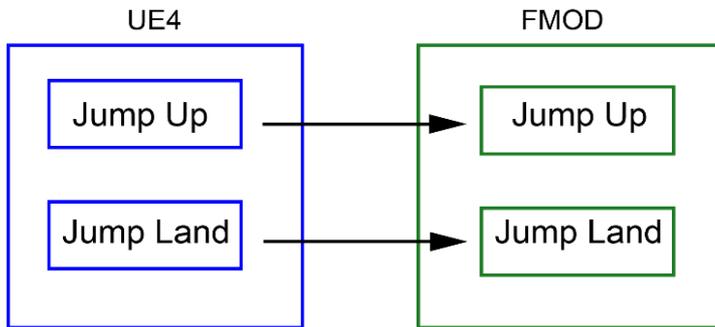


Abbildung 2: Zugriff der UE4 auf FMOD, Ausgangssituation

Wenn der Sound Designer sich dazu entscheidet, beides in einem Event `Jumping` umzusetzen – mit den Parametern `JumpUp` und `JumpLand` – und die Programmierung nicht in Kenntnis gesetzt wird, werden zwei nicht existente Events durch die nicht entsprechend angepassten Methoden gerufen. Als Folge hiervon wird keins der beiden Geräusch abgespielt (vgl. Abbildung 3).

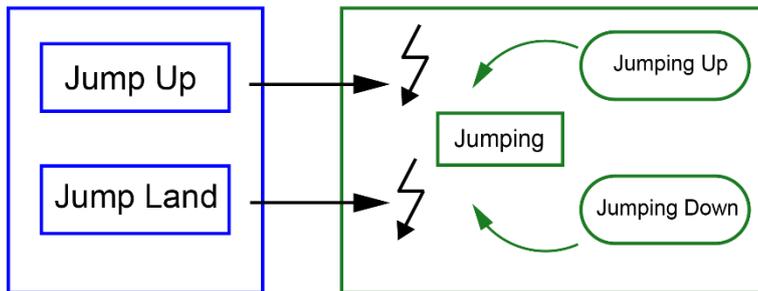


Abbildung 3: Zugriff der UE4 auf FMOD, Fehler

In der Engine müssen die Methoden dem FMOD-Event entsprechend angepasst werden; dies kann beispielsweise so aussehen, dass ebenfalls nur eine Methode `Jumping` mit den Parametern `JumpUp` und `JumpLand` vorliegt (vgl. Abbildung 4).

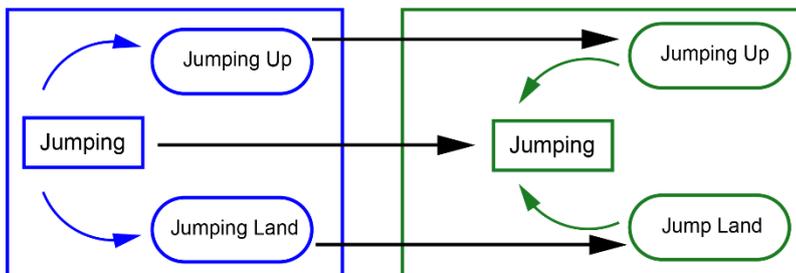


Abbildung 4: Zugriff der UE4 auf FMOD, Korrigierte Version

4.1.4 Technische Rahmenbedingungen

Nach den Gesprächen mit den verschiedenen Departments, sollten nun einige Dinge geklärt sein. Ein wichtiger Aspekt sind dabei die technischen Rahmenbedingungen.

Auch wenn die Computerleistung von heute enorm ist, im Vergleich zur Pionierzeit der Games, so ist der Speicher, welcher der Vertonung zugeschrieben wird, dennoch relativ stark beschränkt. Dies geht darauf zurück, dass mit dem Speicherbudget auch alle Elemente eines Games, also die Objektanzahl, die Auflösung, die Größe der Spielwelt und auch die Erwartungen der Konsumenten gegenüber allen Aspekten eines Computerspiels gewachsen sind (Vachon, 2009). „Das Spannungsverhältnis zwischen Speicherplatzaufwand auf der einen und Klangqualität und Variantenreichtum auf der anderen Seite ist eines der wichtigsten Kriterien für die Erstellung von Ton und Musik für Videospiele[...].“ (Leenders, 2012, S. 32) Es ist auch anzumerken, dass sich die Situation für die Vertonung mittlerweile gebessert hat und ihre Wichtigkeit zumindest in Grundzügen anerkannt wird. Es hängt also davon ab, was das Alleinstellungsmerkmal des Spiels sein soll; je realistischer beispielsweise das Aussehen sein soll, desto weniger Speicher wird für die Vertonung übrig bleiben. Ist jedoch eine realistische Vertonung mit Mehrkanalton angestrebt, so wird selbstverständlich auch der Vertonung mehr Kapazität zugeschrieben (Marks, 2017).

Im engen Zusammenhang dazu steht die angestrebte Plattform¹⁶, da die verfügbaren Ressourcen an die Möglichkeiten eben jener gebunden sind. Unter der Prämisse, dass sich die Hardware ständig weiterentwickelt, lässt sich im Vergleich von Computer und Konsole sagen, dass die Ressourcen in der Regel bei Konsolen strikter begrenzt sind. Dies ist auf die einheitliche Hardwareausstattung zurückzuführen, die sich auch nur bedingt erweitern lässt. Die PlayStation 4 verfügt etwa über einen achtkernigen 1,6 GHz Prozessor, 8 GB Ram, eine 500 GB Festplatte, sowie einen eigenen Audio-Prozessor und ist in der Lage, ca. 200 Kanäle simultan zu spielen; die Xbox One auf der anderen Seite besitzt eine 1,75 GHz Achtkern-CPU, 8 GB Ram, 500 GB Festplatte, ebenfalls ein eigenen Audio-Prozessor und kann ca. 512 Kanäle gleichzeitig abspielen (Marks, 2017). Dem gegenüber steht die Vielfalt an Computern; Hochleistungs-Gaming-PCs sind ebenso die Realität wie leistungsschwächere Office-Laptops. Es gilt also abzuwägen: Auf Konsolen stehen in der Regel weniger Ressourcen zur Verfügung, dafür ist gewährleistet, dass das Spiel auf jeder Konsole gleich gut läuft; dagegen sind für Computer potentiell ressourcenintensivere Spiele möglich, mit der Gefahr, dass diese auf einigen Rechnern wegen zu schwacher Hardware nicht spielbar sind. Zwar sind unterschiedliche Grafikeinstellungen häufig möglich, damit Spiele auch auf weniger leistungsstarken Rechnern laufen, jedoch wird bei manchen Spielen vom Konsumenten erwartet, im Besitz einer entsprechenden Hardware zu sein.

¹⁶ Unter *Plattform* wird das verwendete elektronische *System* bezeichnet, welches zum Spielen genutzt wird. Darunter fallen alle Spielkonsolen, Computer und Handheld-Devices (Horowitz & Looney, 2014).

Es muss bedacht werden, dass anders als im Film, sämtliche Sound-Dateien separat vorliegen, der Speicherbedarf also größer ist. Direkt vom Hardwarebudget ist die Klangqualität, die mögliche Varianz als auch die maximale Zahl an gleichzeitigen Klängen abhängig (Leenders, 2012). Nutzt ein Game das Streaming-Verfahren, so wird – um Speicherplatz und Prozessorleistung zu sparen – auf in der Regel komprimiertes Audiomaterial zurückgegriffen. Dieses wird dann nach und nach von der Festplatte geladen und dekomprimiert, im RAM gepuffert und abgespielt (Marks, 2017).

Wird eine Middleware genutzt, so wird beim Builden der Banks das Audiomaterial in ein wählbares Format konvertiert und, falls ausgewählt, auch komprimiert. Das bedeutet, der Tonschaffende kann bis zu diesem Schritt mit einer beliebig hohen *Samplerate*¹⁷ und *Bittiefe*¹⁸ arbeiten (es wird an dieser Stelle ein von der Middleware lesbares Format vorausgesetzt). Um eine durchgängig gute Tonqualität zu gewährleisten, sollte mindestens mit der aus der Musik üblichen Sample-rate von 44.1 kHz oder den im Filmbereich und TV verwendeten 48 kHz und Bittiefen von 16, 24 oder 32 Bit zu gearbeitet werden, auch wenn später in der Regel auf 16 Bit herunterkonvertiert wird. Selbst wenn in einem Computerspiel unkomprimiertes PCM Sound Design und Musik (z.B. .wav oder .aiff) verwendet wird, ist der Mehrwert durch höhere Sampleraten, wie etwa 192 kHz nur begrenzt, da der durchschnittliche Hörer kaum in der Lage ist, Unterschiede zu hören. Zudem können bisweilen auch größere Datenmengen im Verlauf eines Projekts zusammenkommen, da, wie bereits mehrfach erwähnt, alle Sounds einzeln vorliegen müssen (Marks, 2017). Es ist zumal nicht selbstverständlich, dass das Endgerät höhere Sampleraten als 48 kHz auch abspielen kann. Es bietet sich daher aktuell (Stand 2017) eher an, möglichen Speicherplatz für eine höhere Varianz der Sounds und/oder Musik zu investieren, als für höherer Sampleraten, da die Varianz der Immersion und dem Spielerlebnis in größeren Sprüngen zuträglich ist. Dies kann sich in Zukunft noch ändern.

Bekannt sollte nun auch sein, welches Kanal-, beziehungsweise Wiedergabesystem verwendet wird, sprich ob das Endresultat in Mono, Stereo, Surround oder 3D-Audio erlebt werden soll.

Survival-Spiele verfügen in der Regel über verhältnismäßig wenig Dialog, da der Plot des Spiels, gemessen an dem anderer verwandter Genres, eher nebensächlich ist. Während Shooter nicht für ihre Dialoge bekannt sind, bestand selbst der Dialog von HALO (2001) aus etwa sechstausend

¹⁷ Die Samplerate, oder Abtastfrequenz beschreibt wie viele Abtastwerte dem abzutastenden Signal pro Sekunde entnommen werden. Bei einer Samplerate von 44.1 kHz werden 44.100 Samples in der Sekunde entnommen (Dickreiter, 2014).

¹⁸ Die Bittiefe repräsentiert die möglichen Werte der Amplitude, die ein Sample während der Quantisierung annehmen kann. Sie stellt also gewissermaßen die Abstufung und dadurch den möglichen Dynamikumfang dar; 1 Bit entspricht dabei etwa 6dB. Demnach hat ein Signal mit 24 Bit einen Dynamikumfang von 144 dB (Dickreiter, 2014).

Zeilen (O'Donnell, 2002), wohingegen der Survival-Titel THE FOREST (2014) so gesehen mehr oder weniger völlig auf Dialog verzichtet. Ebenso der Bedarf an komplexer Musik ist vergleichsweise gering, da durch eine simple atmosphärische Musik das Gefühl des Alleinseins unterstrichen und der Spieler, trotz erlebter Extremsituationen im Spiel nicht überfordert wird; ein gutes Beispiel hierfür ist THE LONG DARK (2014). Die hierdurch freiwerdende Kapazität kann für eine größere Varianz der Sounds verwendet werden.

THE DARK SIDE OF THE MOON beispielsweise ist, wie diverse andere Survival-Spiele als reines PC-Game konzipiert. Die Menge an Dialog ist relativ überschaubar; die meisten Assets dieser Art sind Warnhinweise auf kritische Charakterzustände. Die Musik ist ebenfalls minimalistisch gehalten. Da zusätzlich das Spiel in Levels unterteilt ist, die separat geladen werden, stand der Vertonung keine definierte Grenze, solange sie nicht zu Performanceeinbußen führte; die nicht-komprimierten Assets (48 kHz, 24 Bit), mit denen in der Middleware gearbeitet wurde, nehmen in etwa 340 MB Festplattenspeicher in Anspruch. Der Output liegt als 5.1 Surround vor und die Files sind im Vorbis-Format (komprimiert) gespeichert.

4.1.5 Was benötigt alles eine Vertonung?

Unter der Prämisse, als Audio-Director und Sound Designer direkt von Anfang an in einer Produktion beteiligt zu sein, oder das es aus anderen Gründen keine vorgefertigte Audio-Asset-Liste gibt, gilt es bevor die Sounds erschaffen werden, eine solche anzufertigen. Dazu muss evaluiert werden, was alles im Spiel einer Vertonung bedarf. Um dies zu klären, gibt Cullen (2016) eine Checkliste an, mit der man die Notwendigkeit von Sounds herausarbeiten kann:

- Kann sich das Objekt bewegen? Wenn ja, so erwartet das menschliche Bewusstsein ein Geräusch.
- Zieht es Aufmerksamkeit auf sich? Wenn ja, soll dieser Effekt durch eine Vertonung verstärkt werden?
- Soll dem Spieler etwas mitgeteilt werden, was er nicht sehen kann, beziehungsweise außerhalb seines Blickfeldes ist? Durch die Natur von Klängen sind sie rundum wahrnehmbar.
- Soll eine Stimmung erregt werden? Klänge berühren immer auch emotional.

Zusammen mit der theoretischen Grundlage sollte die Erstellung der Asset-Liste nun erleichtert werden. Aus den Auditory Icons geht beispielsweise hervor, dass ein Objekt, mit dem interagiert

werden kann, deshalb einen Sound benötigt, da dem Spieler ansonsten eine Rückmeldung fehlt, ob seine Eingabe angenommen und korrekt umgesetzt wurde. Aus dem Wissen, dass Geräusche den Hörer über materielle Beschaffenheit informieren leitet sich ab, dass unterschiedliche Böden jeweils unterschiedliche Fußschrittgeräusche benötigen, da ein unpassender Sound direkt erkannt und dadurch die Immersion geschmälert wird.

Nicht direkt Teil der benötigten Sounds, aber auch wichtig im Kontext der Erstellung der Asset-Liste ist die Einteilung der im Spiel vorkommenden Räume nach ihren klanglichen Merkmalen. Soll der Raumklang dieser mit einem *Faltungshall*¹⁹ unterstützt werden, sind auch möglicherweise benötigte *Impulsantworten*²⁰ Bestandteil der Liste.

Dabei sollte die Nomenklatur der Assets eine eindeutige Identifizierung erlauben. Hier ein Auszug der Asset-Liste von THE DARK SIDE OF THE MOON (Wagner, 2017), in der die benötigten Assets für die Möbel aufgelistet sind:

<i>Asset name</i>	<i>Category</i>	<i>Description</i>
<i>glass_breaking</i>	<i>furniture, destruction</i>	<i>breaking glass (smashing)</i>
<i>wood_crack</i>	<i>furniture, destruction</i>	<i>cracking wood</i>
<i>wood_impact_a</i>	<i>furniture, destruction</i>	<i>hitting wood (with a crowbar)</i>
<i>wood_impact_b</i>	<i>furniture, destruction</i>	<i>low-frequency component to hitting wood</i>
<i>wood_impact_c</i>	<i>furniture, destruction</i>	<i>high-frequency component</i>
<i>locker_close</i>	<i>furniture, metal</i>	<i>shutting a locker</i>
<i>locker_open</i>	<i>furniture, metal</i>	<i>opening a locker</i>
<i>chest_open</i>	<i>furniture, wood</i>	<i>chest snapping open (sound like a suitcase?)</i>
<i>chest_close</i>	<i>furniture, wood</i>	<i>closing the chest (also like a suitcase?)</i>
<i>chest_electric_shock</i>	<i>furniture, wood</i>	<i>player receives an electric shock after failed lock picking attempt</i>
<i>chest_selfdestruction</i>	<i>furniture, wood</i>	<i>after 3 failed lock picking attempts, the chest will destroy what's inside</i>
<i>closet_close</i>	<i>furniture, wood</i>	<i>shutting the door of a closet (will also be used for night table)</i>

¹⁹ Ein Faltungshall verrechnet, genauer faltet, das Audiosignal mit einer Impulsantwort. Das resultierende Signal entspricht dann dem Signal, welches in einem realen Raum an der jeweiligen Position und dem entsprechendem Equipment eingefangen wurde (Dickreiter, 2014).

²⁰ Eine Impulsantwort beschreibt das zeitliche Verhalten eines Systems, nach dessen Anregung durch einen Impuls (Dickreiter, 2014).

<i>closet_open</i>	<i>furniture, wood</i>	<i>opening the door of a closet (will also be used for night table)</i>
<i>drawer_close</i>	<i>furniture, wood</i>	<i>shutting a drawer</i>
<i>drawer_open</i>	<i>furniture, wood</i>	<i>opening a drawer</i>

In dieser Asses-Liste sind nur die benötigten Sounds benannt, nicht wie viele Variationen davon später angefertigt worden sind.

4.1.6 Akustische Perspektive

Was hört der Spieler? Und vor allem wie hört er es? Die *akustische Perspektive* ist eine Frage sowohl bezüglich technischer, als auch gestalterischer oder inhaltlicher Gesichtspunkte.

Auf inhaltlich-gestalterischer Seite lässt sich die akustische Perspektive durch die *Fokalisierung* beschrieben, ein aus der Literatur stammendes Konzept, das von Flückinger auf den Film übertragen wurde und sich wiederum auf Computerspiele erweitern lässt. Diese Übertragung ist deshalb möglich, da es sich „bei der Fokalisierung um einen perzeptiven Standpunkt handelt. [...] In Fragen übersetzt, heißt es: Wer erzählt, wer hört, wer sieht?“ (Flückiger, 2002, S. 372 f.)

Die drei grundlegenden Unterteilungen sind (Flückiger, 2002):

a) Null-Fokalisierung:

Die Null-Fokalisierung entspricht einem auktorialen/allwissendem Erzähler der Literatur. Der Spieler hat einen gottgleichen Blick auf das Geschehen, ist dabei jedoch kein direkter Bestandteil der Spielwelt. Diese Perspektive ist also nichtdiegetisch. Ein Beispiel hierfür wären etwa Strategiespiele, sowohl RTS wie etwa die AGE OF EMPIRES Reihe, als auch TBS wie die TOTAL WAR Reihe (wobei angemerkt werden muss, dass eventuell ein *Nebel des Krieges* das Allwissen des Spielers verhindern kann).

b) Interne Fokalisierung:

Das Spielgeschehen wird im Falle einer *internen Fokalisierung* aus der Sicht des Spielcharakters wahrgenommen. Der POV in Computerspielen unterscheidet sich dahingehend von einem POV-Shot im Film, dass nicht nur ein kurzer Ausschnitt aus der Sicht, beziehungsweise der Empfindung des Charakters gezeigt, sondern auch die restliche Wahrnehmung des jeweiligen Charakters über die gesamte, oder zumindest eine längere Zeit im Spiel widergespiegelt wird. Es ist eine sehr diegetische Perspektive, die sich im Falle von Computerspielen näher an dem literarischen Vorbild orientiert, als das Verständnis im Film. Dabei muss eine interne Fokalisierung nicht unbedingt aus der Ego-Perspektive sein, auch wenn dies auf den ersten Blick so anmuten mag. Auch ein Spiel

in der Third-Person-Perspektive kann eine interne akustische Perspektive darstellen. In diesem Fall muss sie nicht einmal an die Ohren der Figur gebunden sein, der Grund ist jedoch eher technischer Natur und wird im Laufe des Kapitels erläutert.

c) Externe Fokalisierung:

Die *externe Fokalisierung* könnte als Mischform der Null- und der internen Fokalisierung gesehen werden; die Wahrnehmung ist nicht die einer Person des Geschehens selbst, dennoch Teil der Diegese. Die Motivationen und Gedanken des Charakters bleiben, im Gegensatz zur internen Fokalisierung, verborgen.

Eine Unterteilung dieser Perspektiven ist, wie Flückinger im Falle des Films anmerkt, nicht immer klar und eindeutig (2002). Um dies zu verdeutlichen, ein Computerspiel-Beispiel:

In THE WITCHER 3: WILD HUNT (2015), spielt man überwiegend den ‚Hexer‘ Geralt von Riva (Hexer sind hier eine Art Monsterjäger). Das Spiel ist klar aus der Sicht Geralts und an sich intern fokalisiert; Geralts Gedanken, Empfindungen und Eindrücke sind dem Spieler zugänglich und bekannt. Hexer zeichnen sich im Spiel dadurch aus, über übermenschliche Sinneswahrnehmungen zu verfügen; sie können beispielsweise Lebewesen in relativ großer Distanz anhand ihrer Geräusche wahrnehmen. Der Spieler kann diese Geräusche hören, während NPCs dies nicht können.

Geralt ist auf der Suche nach seiner Ziehtochter Ciri. Er trifft auf dieser Suche immer wieder auf andere Charaktere, die ihm von ihr berichten. Diese Erzählungen sind im Spiel als eine Art Flash-Back umgesetzt, in denen der Spieler in die Rolle von Ciri schlüpft und ihren Charakter spielt. Auch wenn die Hexer-Sinne immer noch anwendbar sind – sie sind ein wichtiger Bestandteil der Spielmechaniken –, erfährt der Spieler technisch gesehen nicht direkt was Ciri empfindet und erlebt; alles basiert eigentlich nur auf den Erzählungen von NPCs. Da diese NPCs wiederum Teil der Spielwelt sind, können diese Spielabschnitte als eine externe Fokalisierung verstanden werden. So vereint THE WITCHER 3: WILD HUNT (2015) mindestens zwei Arten der Fokalisierung.

Um dem Spieler auch eine Orientierung und Ortung zu ermöglichen, ist es zusätzlich notwendig, den eigentlichen Ort der Schallaufnahme zu definieren. Dies geschieht mittels eines sogenannten *Listeners* in der Game Engine. Dieser Listener repräsentiert im übertragenen Sinne die Ohren des Spielers in der Spielwelt. Da er stets eine definierte Position innerhalb der Spielumgebung besitzt, kann diese mit der Position möglicher Schallquellen verglichen und so ein Hörereignis berechnet werden (Leenders, 2012).

Standardmäßig ist dieser Listener an die Kamera fixiert. Im Falle eines Spiels aus der Ich-Perspektive ist dies auch durchaus sinnvoll und richtig. Im Falle einer Third-Person- oder einer Top-

Down-Perspektive ist dies nicht unbedingt korrekt. Ist die Kamera beispielsweise sehr weit über dem Geschehen, so ist es möglich, dass durch die zu große Entfernung zwischen Listener und Schallquelle, das resultierende Schallereignis viel zu leise, oder im schlimmsten Fall gar nicht hörbar ist (Sykora, 2016).

Es ist in diesem Fall also unter Umständen sinnvoller, den Listener an den Charakter zu binden, oder aber seine Position an die Kopfposition anzupassen. Beide Varianten führen jeweils zu einem unterschiedlichen Resultat:

- a) Der Listener ist fest an den Charakter gebunden, etwa seinen Kopf. Das akustische Abbild, das der Listeners zeichnet, entspricht der theoretischen Hörumgebung der Spielfigur. Die Richtungslokalisierung von Schallquellen ändert sich bei einem rotieren der Kamera nicht mit, ihr Panning wird nicht angepasst.
- b) Der Listener ist an die Kamera gebunden, dafür um einen Offset dermaßen versetzt, dass er mit der Charakterposition zusammenfällt. Wird die Kamera geschwenkt, so ändert sich auch die Richtungslokalisierung entsprechend; das Panning der Schallquellen wird angepasst.

Ein Beispiel hierzu: Das Kind rechts im Vordergrund in Abbildung 5 und auch der Reiter hinter diesem, sind bei einer Rückansicht des Charakters (weißhaariger Mann in der Bildmitte) jeweils von rechts zu hören. Schwenkt der Spieler die Kamera um 180 Grad, so sind beide im Fall a) immer noch von rechts zu hören, im Fall b) von links. a) stellt die realistischere Variante in Bezug auf das Hörerlebnis des Charakters dar, kann jedoch für Verwirrung beim Spieler sorgen, da in diesem Fall das Gesehene nicht mehr mit dem Gehörten übereinstimmt.



Abbildung 5: Spielszene aus THE WITCHER 3: WILD HUNT (Thudium, 2015)

Die Kombination aus Fokalisierung und der Position des Listeners entscheiden, wie die Spielwelt wahrgenommen wird. Es ist also wichtig abzuklären, ob die Klänge objektiv oder durch den Charakter beeinflusst gehört werden sollen. Die Position des Listeners kann helfen, die notwendige Größe des ‚Hörfelds‘ zu definieren; zudem kann das Verständnis der Beziehungen der Listener-Position spätere Problemquellen frühzeitig aufdecken.

THE DARK SIDE OF THE MOON wird vollständig aus der Sicht von Robert Goodman gespielt, sowohl visuell, als auch aural und emotional. Der Listener ist, da es aus der Ego-Perspektive gespielt wird, an die Kamera geheftet und entspricht dem Höreindruck Robs. Zudem ist die gesamte Vertonung darauf ausgelegt, eine interne Fokalisierung, im Sinne seiner Einsamkeit und Verlorenheit auszudrücken. Von lauten Fußschritten, über das laute Denken Robs, bis hin zur melancholisch-atmosphärischen Musik.

4.2 Sound Design

Beim Sound Design für Computerspiele können so ziemlich alle Tricks und Kniffe zur Erstellung von Sounds aus der Filmwelt übernommen werden, da sich die Produktion nur marginal unterscheidet. Vielmehr sind es die Arbeitsschritte ab dem Edit, welche sich unterscheiden (Leenders, 2012).

Die eigentliche Kreation der Sounds beinhaltet insbesondere die vier Gebiete: Arbeit mit Sound-Libraries, Studioaufnahmen, Field-Recordings und Nachbearbeitung. Das Prinzip des Layering kann dabei gewissermaßen jeden der Schritte begleiten, ist jedoch eigentlich ein Bestandteil der Nachbearbeitung.

Wichtig für die Umsetzung der Sounds, ist ein Verständnis der Abnutzung und der Varianz.

4.2.1 Varianz und Abnutzung

Das große Ziel eines Computerspiels ist es, den Spieler völlig in das Spielgeschehen zu entführen. Jeder kleine Fehltritt ist in der Lage, die Immersion des Spiels zu gefährden. Während Spieler möglicherweise nicht in der Lage sind, handwerkliche oder technische Fehler einer Vertonung zu hören und/oder zu benennen, wie etwa ein unbeabsichtigtes Pumpen durch zu starke Kompression, oder eine leichte Verzerrung, so fallen häufig auftretende Wiederholungen selbst einem Laien sofort negativ auf (Vachon, 2009).

Wiederholungen, im schlimmsten Falle innerhalb einer kurzen Zeitspanne, können nicht vermieden werden. Gründe hierfür sind unter anderem:

- Der nichtlineare Spielablauf:
Es ist nicht vorhersehbar, wann der Spieler wohin geht. Ein Beispiel: Gegeben sein die Gebiete Haus und Straße, mit jeweils einer dazugehörigen Musik. Der Spieler tritt von der Straße in das Haus, bemerkt, dass er sich nicht am gewünschten Ort befindet und verlässt es sogleich wieder. Er befindet sich also wieder auf der Straße. Die Musik ändert sich also von Straße zu Haus und zurück. Insofern die Musik jedes Mal von vorne beginnt, wird der Spieler den Anfang der Stücke potentiell überdurchschnittlich oft hören. Dies gilt analog auch für Atmos.
- Die Natur von Computerspielen:
Bestimmte Aktionen müssen wiederholt ausgeführt werden. Bedingt werden diese Aktionen durch die Spielmechaniken, beziehungsweise können diese Aktionen die Mechaniken verkörpern. In Survival-Games müssen beispielsweise die lebenserhaltenden Bedürfnisse in nicht allzu großen Abständen befriedigt werden. Es muss immer Ausschau nach Lebensmitteln gehalten, diese gesammelt und anschließend verzehrt werden.
- Endlichkeit der Hardwarekapazität:
Es ist nicht genügend Speicherplatz vorhanden, um jeden einzelnen Sound oder ein Musikstück wirklich nur ein einziges Mal in einem Spiel verwenden zu können. Hierzu kann die Musik als veranschaulichendes Beispiel zurate gezogen werden. Eine Komposition für ein dreißig- bis vierzigstündiges Spielerlebnis besteht in der Regel aus etwa sechzig bis neunzig Minuten Musik (Rumsey, 2010).

Erkennt der Spieler diese Wiederholung, ist es nur eine Frage der Zeit, bis er von ihnen genervt ist. Die Vertonung leidet unter einer *Abnutzung* (Leenders, 2012). Alle Kategorien einer Computerspielvertonung – Sound Design, Dialog und Musik – können unter der Abnutzung durch Wiederholungen leiden. Jedoch müssen diese jeweils gesondert voneinander betrachtet werden, da sie sich für eine universelle Lösung zu stark voneinander unterscheiden (Vachon, 2009).

Als erstes soll das Sound Design betrachtet werden; dazu ist ein kurzer Exkurs in einen anderen Bereich angebracht: Eine ähnliche Problematik ist im Bereich des Samplings von Instrumenten als *Machine-Gun-Effect* bekannt; durch die monotone, rasche Abfolge zum Beispiel eines Kick-Drum Samples in derselben Anschlagstärke, entsteht im metaphorischen Sinne der Eindruck eines Maschinengewehrfeuers. Um dem entgegenzuwirken, wird auf alternierende Samples gesetzt. Diese *Round-Robin-Samples*, sind verschiedene Aufnahmen von Schlägen derselben Stärke. Umso größer die Anzahl dieser Samples ist, oder auch bei einer randomisierten Reihenfolge der

Samples, sodass nicht zweimal hintereinander der gleiche Schlag zu hören ist, desto echter klingt das Instrument (Obst, kein Datum).

Denselben Ansatz verfolgt das Anlegen einer *Varianz*; es werden also Sound FX derselben Art in einer zufälligen Reihenfolge getriggert. Diese können zusätzlich mit Effekten versehen werden (Leenders, 2012). Diese Effekte sollten dabei jedoch nicht blind oder übermäßig eingesetzt werden. Beim Einsatz von zufälligem Pitch-Shifting sind beispielsweise das veränderte Timbre und ein möglicherweise unnatürlicher Klang zu beachten. Zudem ist der alleinige Einsatz Pitch²¹- oder Volume-Randomization²² ungeeignet, da schnell auffällt, dass derselbe Sound in höher/tiefer beziehungsweise lauter/leiser abgespielt wird (Vachon, 2009).

Damit die Varianz funktionieren kann, müssen die Sounds paradoxerweise sehr ähnlich im Klang sein. Einzelne Sounds, die zu sehr aus dem auralen Muster fallen, ziehen ansonsten besonders Aufmerksamkeit auf sich und fallen beim erneuten Vorkommen auf (Vachon, 2009); die Immersion ist in diesem Fall gestört.

Weitere mögliche Lösungen, um einer Abnutzung entgegenzuwirken sind das Echtzeit-Layering und auch ein vergrößerter zeitlicher Abstand zwischen dem Abspielen eines Sounds (Leenders, 2012). Dies ist auch eine Erklärung dafür, warum beispielsweise die Quest-Abschluss-Sounds in *THE WITCHER 3: WILD HUNT* (2015) kaum unter einer Abnutzung leiden. Da für das Erledigen der Quests zum Teil viel gereist werden muss und selten mehrere Quests auf einmal ihr Ende nehmen, vergeht zwischen ihnen eine längere Zeitspanne; diese kann sich von ein paar Minuten bis hin zu Stunden spannen, je nach Spielerverhalten.

Sowohl Leenders (2012), als auch Vachon (2009) vor ihm, weisen darauf hin, dass eine Wiederholung von zum Beispiel HUD- oder Menü-Sounds, beziehungsweise auch beim Aufsammeln von Items wichtig sei. Leenders (2012, S. 41 f.) nennt als Begründung dieser These die Wichtigkeit der „Bestätigungsfunktion der Sounds“. Diese sei gefährdet, insofern der verwendete Sound nicht jedes Mal derselbe sei. Vachon (2009) dagegen berichtet von negativen Reaktionen aufgrund verschiedener Sounds in diesem Kontext, seitens des Entwicklungsteams

Zustimmung ist im Falle der nichtdiegetischen Sounds von zum Beispiel Menü oder auch dem zuvor angesprochenen Beispiel der Quest-Abschlüssen auszusprechen. Durch Playtests²³ von *THE DARK SIDE OF THE MOON* war jedoch in Bezug auf das Aufsammeln von Objekten etwa, oder

²¹ Pitch-Randomization: zufällige Tonhöhenveränderung

²²Volume-Randomization: zufällige Amplitudenänderung

²³ Bei den Tests handelte es sich um ein heuristisches Verfahren mit nur wenigen Teilnehmern. Demnach müssten diese Erkenntnisse durch weitere Tests überprüft und gesichert werden um eine wissenschaftlich fundierte Aussage zu treffen. Es ist anzumerken, dass sich nichts desto trotz Schlussfolgerungen aus solchen Tests ziehen lassen und mögliche Trends aufgedeckt werden können.

auch der Konsum derer ein anderer Trend zu beobachten. Es fanden Tests in zwei Phasen der Vertonung statt: in der ersten Phase waren nur einzelne Sounds ohne Varianz als Platzhalter implementiert. Im zweiten Test waren bereits sämtliche Variationen integriert. Die bestätigende Funktion blieb erhalten, da die Sounds von eben diesen Aktionen, also für das Aufnehmen, Crafting, Essen, Trinken, etc. insgesamt sehr ähnlicher Natur waren und aus denselben Komponenten bestanden. Die Tester empfanden diese Version der Tests als angenehmer.

In Survival-Games, verglichen mit Spielen anderer Genres, treten jene Sounds extrem oft auf, da sie die Kern-Spielmechaniken repräsentieren; die jeweils den Sounds zugehörigen Aktionen müssen in recht kurzer Zeit mehrfach und andauernd ausgeführt werden, ansonsten stirbt der Spieler den Bildschirmtod. Es ist also nicht verwunderlich, dass in andern Genres der positive Effekt der Wiedererkennbarkeit und der Bestätigungsfunktion überwiegen; die größeren Intervalle zwischen dem Auftreten dieser Geräusche wirken einer raschen Abnutzung entgegen. Survival-Spiele nehmen in Anbetracht dessen also eine Sonderposition ein.

Auch wenn Dialog eigentlich nicht zum Sound Design gehört, muss im Zusammenhang der Abnutzung dennoch gesagt sein, dass die Wiederholung von Dialog eine der am schnellsten negativ wirkenden Arten von Audio-Wiederholung darstellt. Da Menschen sich nie zweimal genau gleich ausdrücken – Artikulation, Betonung und Tempo sind nur ein paar Merkmale – werden mehrfach gehörte Dialogzeilen sofort entlarvt. Dabei stellen Story-Dialoge in der Regel kein Problem dar, da sie eigentlich ohnehin nur einmal auftauchen. Problematisch sind sogenannte *Barks* oder auch *Vocal Sound Effects*; dies sind alle nicht-story Dialoge, inklusive zufälliger Sprüche und Ausrufe (Vachon, 2009). Das Problem hierbei ist, dass die Möglichkeiten der unterschiedlichen Dialogvariationen, wie bereits besprochen, direkt vom verfügbaren Speicherplatz abhängen. Eine Möglichkeit dennoch Abwechslung zu schaffen, kann ein ‚Stückelungssystem‘ sein, in welchem diverse Satz- oder Spruchbestandteile zufällig kombiniert werden können. Dabei ist auf einen konstanten Klang in jeder Konstellation zu achten, was seinerseits wiederum einen hohen Aufwand darstellt. Seit längerem findet diese Methode in Sportspielen in Gestalt der Voice-Over der Moderatoren Verwendung (Vachon, 2009), wäre jedoch auch auf andere Spielegattungen anwendbar.

In THE DARK SIDE OF THE MOON stellen die statusbezogenen Sprüche eine mögliche Störquelle in diesem Sinne dar. Dennoch sind sie unerlässlich für die Usability, da sie den Spieler über kritische Werte, die Gegenmaßnahmen bedürfen, informieren. Einer übermäßigen Repetition und damit Abnutzung wurde entgegengewirkt, indem mehrere unterschiedliche Sprüche zu jeweils einem Bedürfnis aufgenommen wurden, für Hunger zum Beispiel (Wagner, 2017, S. 6 f.):

Rob

- *I'm starving.*

- *I really could use something to eat.*
- *Just a little snack.*
- *My stomach just started to digest itself.*
- *I'd even eat toilet paper if I'd find some.*

Jen

- *Rob, you should eat something.*
- *You know that humans die if they won't eat for too long, right?*
- *You really need some calories.*
- *Robert Goodman. Eat something. Now!*
- *Martians called. They said they heard your stomach growling.*
- *If you starve, it's your fault.*

Zusätzlich ist das zweite Triggern desselben Events innerhalb einer Minute gesperrt. Den eigentlichen Sprüchen innerhalb des FMOD-Events wurde außerdem eine Wahrscheinlichkeit zugeteilt, sodass nicht bei jedem Auslösen eines Events auch tatsächlich ein Spruch abgespielt wird.

Zwei Möglichkeiten, einer Abnutzung von Musik in Computerspielen entgegenzuwirken, werden im Kapitel 4.4 vorgestellt.

Es ist also wichtig beim Sound Design Prozess darauf zu achten, mögliche Abnutzung zu erkennen und Schritte dagegen zu unternehmen. Im Kontext der Soundkreation kann dies zum Beispiel das Einplanen von Layern bedeuten. Vor allem bedeutet es aber, dass ein einzelner Sound seiner Art in vielen Fällen nicht ausreichend ist und stattdessen direkt eine Zahl an Variationen davon mitproduziert werden sollten.

4.2.2 Sound Libraries

Eine *Sound Library* ist eine Sammlung von Sound FX, die nach thematischen und funktionalen Kategorien eingeteilt sind (z.B.: thematisch – Autos, funktional – Atmo). Dabei liegen die einzelnen Sounds in der Regel bereits editiert, zum Beispiel als One Shot vor. In Games ist die Verwendung von Libraries stärker ausgeprägt als im Film. Dies hat diverse Gründe: Zum einen aufgrund der unterschiedlichen Zeit- und Finanzbudgets und zum anderen aus den unterschiedlichen historischen Entwicklungen von Film- und Computerspielvertonung (Leenders, 2012).

Libraries können insbesondere dann nützlich sein, wenn schwer reproduzierbare Sounds benötigt werden, wie etwa Schussgeräusche von spezifischen Gewehren. Kommerzielle Libraries können

sehr umfangreich sein und sind für gewöhnlich in hoher Audioqualität angelegt; zudem wird mit dem Kauf einer kommerziellen Library in der Regel auch eine Lizenz für die kommerzielle Verwendung dieser Sounds erworben, was bei kostenfreien aus dem Internet nicht unbedingt der Fall ist; zumal auch geschützte Sounds in freien Libraries zu finden sein können. Deren Nutzung könnte unter Umständen Grundlage eines Rechtsstreits werden (Marks, 2017).

Mit Libraries kann potentiell eine große Zahl grundsätzlicher Sounds (gemeint sind in vielen Projekten benötigte Sounds wie etwa Fußschritte) gestemmt werden (Marks, 2017). Dabei sollte jedoch stets im Hinterkopf behalten werden, dass sie durch ihre weite Verfügbarkeit, ihre Einzigartigkeit verlieren. Durch Verfremdung und Kombination unterschiedlicher Sounds innerhalb der DAW, etwa durch Layering, können aus Library-Sounds dennoch originelle und einzigartige neue Sounds geschaffen werden (Leenders, 2012).

Nicht zu unterschätzen ist die Sammlung an Sounds, die durch eigene Aufnahmen in der Karriere eines Sound Designers für andere Projekte zusammenkommt. Es bietet sich an, diese zu sortieren und zu einer eigenen, persönlichen Library zu bündeln, so dass sie als Grundlage wiederverwendet werden können.

Der Umfang großer Libraries kann auch eine lange Zeit des Suchens bedeuten, die nicht unterschätzt werden darf. Es muss also vorher abgewogen werden, ob es nicht vielleicht doch schneller ist, etwa die Charaktergeräusche als Foley umzusetzen; hier wäre eine Einzigartigkeit direkt integriert. Häufig ist es auch so, dass eine Library zwar einen Sound beinhaltet, der in die richtige Richtung geht, aber doch nicht ganz der eine richtige ist (Marks, 2017). Auch sehr spezielle oder abstrakte Sounds können der Grund eigener Aufnahmen sein.

In *THE DARK SIDE OF THE MOON* wurde zwar zum Großteil auf eigene Aufnahmen zurückgegriffen, Elemente die jedoch aus Libraries stammen, sind etwa diverse Gletscher-Knacker, die für das Einfrierens der Raumstation bei Nacht kombiniert wurden, sowie die Explosionen der Selbstzerstörung der Kisten.

4.2.3 Studioaufnahmen

Die Motivation von Studioaufnahmen kann vielfältig sein. Wurden keine befriedigenden und zutreffenden Sounds in Libraries gefunden? Ist es Ziel des Projekts absolut originelle und einzigartige Sounds zu haben? Oder ist es doch zeitökonomischer, einen Teil aufzunehmen statt lange zu suchen?

Der größte Vorteil einer Studioumgebung, gegenüber einer Location des Field-Recordings, besteht insbesondere darin, dass sie relativ kontrolliert und frei von Störgeräuschen ist. Zudem kann

auch mit jedem zugänglichem Equipment wie Mikrofontypen/-modellen aber auch zum Beispiel mit dessen Anwendung, etwa diversen Mikrofonpositionen, experimentiert werden.

Auch die Übersichtlichkeit über das aufgenommene Material ist besser, da potentiell neue Sessions erstellt und benannt, sowie Marker gesetzt werden können. Ebenso kann sofort gelayert und das Ergebnis überprüft werden. Dies ist bei Field-Recordings nur dann der Fall, wenn zusätzliches Equipment mitgeführt wird.

Sowohl Foley, als auch Sound FX können Teil der Studioarbeit des Sound Designs sein. Wie bereits zuvor angerissen, ist klassisches Foley als bildsynchrone Aufnahme der Bewegungssounds der Charaktere, wie etwa Fußschritte und Kleidungsrascheln definiert. Nicht charakterbezogene Sounds gehören demnach eigentlich nicht in diese Definition, werden jedoch häufig fälschlicherweise als Foley bezeichnet. Deren korrekte Bezeichnung lautet stattdessen Sound FX (Ament, 2009; Leenders, 2012). Da in Computerspielen eine bildsynchrone Aufnahme wenn dann nur im Falle von Cutscenes möglich ist, wird der Foley-Begriff lockerer gehandhabt und für die generelle Bezeichnung von Charaktersounds verwendet (Marks, 2017).

Als Konsequenz daraus, rücken Sound FX und Foley, im Kontext einer Computerspielproduktion, in der Art der Erzeugung näher zusammen. Beide werden ohne einen Bildbezug, sozusagen frei performt, da sie ohnehin vermehrt als einzelne Sounds benötigt werden.

Bei der Aufnahme von Sounds jeglicher Art, kann es zu zum Teil skurril wirkenden Szenarien kommen; wenn der Sound Designer zum Beispiel allerhand Obst und Gemüse zerdrückt, zerbricht oder gar zerschlägt. Das Brechen von Sellerie und Möhren etwa, ist ein beliebter Trick um brechende Knochen dramatischer zu gestalten (CBS Sunday Morning, 2015). Der Industriestandard für Fußschritte in Schnee, ist das Drücken von Maisstärke in einem Karton oder einem kleinen Säckchen (KPBS News, 2008).

Bei der Umsetzung von Sound FX sollte stets auch darauf geachtet werden, dass weder Mensch noch Equipment im Eifer des Gefechts zu Schaden kommen. Natürlich ist dies immer auch vom jeweilig gesuchten Sound abhängig: Während die Wahrscheinlichkeit, dass Equipment beim Aufnehmen von einfachen Schritten kaputt geht und sich der Foley-Artist verletzt, recht gering ist, besteht zum Beispiel beim Schwingen diverser Gegenstände für Swooshes, durchaus die Gefahr hierfür. Auch Experimente die Schmutz und Feuchtigkeit beinhalten, können Risiken für das Equipment mit sich bringen und unter Umständen zu einer Rutschgefahr führen.

In *THE DARK SIDE OF THE MOON* wurde das gesamte Foley im Studio kreiert. Dabei wurden zum Beispiel Fußschritte, verschiedene Kleidungs- und Rucksackraschelgeräusche jeweils separat aufgenommen um sie später in der Middleware in Echtzeit zu layern. Als Repräsentanten der

Sound FX wurden beispielsweise die Türen der Raumstation und das Lagerfeuer im Studio geschaffen. Das Feuer wurde durch diverses Foliennistern, direktes Pusten auf die Mikrofonmembran und das Zerplatzen von Luftpolsterfolie umgesetzt.

Abschließend ist noch anzumerken, dass es sinnvoll ist, Mikrofone und Vorverstärker so zu wählen, dass die resultierenden Aufnahmen möglichst rauscharm sind. Durch das Layering zum Beispiel, kann sich dieses Rauschen ansonsten aufsummieren. An dieser Stelle wird nicht genauer auf die Funktionsweise der verschiedenen Mikrofontypen eingegangen (es wird von einem vorhandenen Hintergrundwissen ausgegangen); so viel sei jedoch gesagt: Für sehr leise Geräusche können die empfindlicheren Kondensatormikrofone eine bessere Wahl sein, insofern ein störungsfreie Umgebung vorherrscht. Zusätzlich sollte darauf geachtet werden, dass stets auch ‚Stille‘ mitaufgenommen wird, da dies bei einem eventuell notwendigen De-Noising benötigt ist.

4.2.4 Field-Recording

Sind Sounds für eine Studioaufnahme unpraktisch oder schlichtweg nicht durch Aufnahmen im Studio realisierbar, wie etwa Fabrikmaschinen oder Schusswaffen, so ist die Zeit für Field-Recordings gekommen. Unter Field-Recordings werden alle Aufnahmen außerhalb eines kontrollierten Studioumfelds verstanden (Marks, 2017). Das gesuchte Klangbild wird also vor Ort aufgenommen.

Zwar ist Planung in der Vertonung eines Computerspiels generell das A und O, doch insbesondere im Falle von Field-Recordings ist eine sorgfältige Planung unerlässlich. Es muss vorher unbedingt geklärt sein, welche Aufnahme, von welcher Schallquelle unter welchen Aufnahmebedingungen aufgenommen werden muss, da diese Sessions häufig eine einmalige Angelegenheit darstellen oder nur durch viel Geld wiederholbar sind (Marks, 2017). Ist nur eine einfache Wald-Atmo eines nahegelegenen Waldes fehlerhaft, so ist dies zwar ärgerlich, aber verhältnismäßig unproblematisch. Sollen aber etwa eine Vielzahl an Gewehrsounds, inklusive Schüsse, unterschiedlichster historischer Gewehre aufgezeichnet werden, etwa für einen Shooter im zweiten Weltkrieg, ist damit zu rechnen, dass sowohl die Schützen bezahlt, sowie die Waffen und der Schießstand angemietet werden müssen. Fehlen hier im Nachhinein einzelne wichtige Sounds, etwa jeweils das Repetieren der Gewehre, ist dies nicht nur ärgerlich, sondern erfordert eine erneute Anmietung und einen unter Umständen hohen logistischen Aufwand.

Für Field-Recordings ist ein mobil einsetzbares Equipment notwendig. Dazu gehören Fieldrecorder, Mikrofone und diverses Zubehör. Die Anforderungen an das Equipment unterscheiden sich zum Teil von den Bedürfnissen im Studio. Die Gegebenheiten der Aufnahmesituation können die Auswahl stark beeinflussen. In einer bestimmten Situation mag vielleicht das Mikrofon, welches

an sich den detailliertesten Klang liefern würde, nicht praktikabel sein, etwa weil die Umgebungsgeräusche ebenfalls stark aufgezeichnet werden, oder weil es nicht robust genug für diese Situation ist (Marks, 2017).

Wichtige Gesichtspunkte des Equipments sind deren Gewicht und Handlichkeit; unter Umständen können Field-Recordings über einen langen Zeitraum gehen, in denen das Equipment in irgendeiner Form getragen werden muss. Ist also etwa ein langer Tag in unwegsamem Gelände geplant, muss das Equipment auch unter diesen Umständen transportierbar sein. Ebenso wichtig ist eine der Situation angepasste Robustheit. Diese kann sich sowohl auf Erschütterungen, Verschmutzung, hohen oder niedrigen Temperaturen und auch Feuchtigkeit beziehen (Marks, 2017). Das benötigte Format hat ebenso einen Einfluss auf das Equipment. Soll eine Atmo etwa in 5.1 Surround vorliegen, und die Kanäle dabei diskret aufgenommen werden, so werden fünf Mikrofone (ohne LFE), sowie eine Möglichkeit alle simultan aufzunehmen, benötigt.

Im Falle von Mikrofonen sind die wichtigsten Fragen, die vor allen anderen Überlegungen zu stellen sind: ist das jeweilige Mikrofon dazu geeignet, den geplanten Sound einzufangen und klingt das Ergebnis gut? Das Mikrofonarsenal kann einiges umfassen, zum Beispiel unter anderem Allzweckmikrofone (das bekannte Shure SM 57 oder SM 58 etwa), Mikrofone für spezielle Anwendungen wie etwa Kontaktmikrofone und stark gerichtete Mikrofone (Marks, 2017).

Bei Mikrofonen sind unter anderem die Wandler-Prinzipie und die Empfängerart wichtig. Daraus lassen sich beispielsweise eine Anfälligkeit gegenüber Wind und Trittschall ableiten²⁴, genauso ob eine Phantomspannung anliegen muss oder nicht²⁵. Auch über die Richtwirkung lassen sich Rückschlüsse ziehen.

Sowohl im Studio, als auch im freien Feld ist das Ziel, das Nutzsignal so deutlich wie möglich einzufangen. Während im Studio von einer geräuscharmen Umgebung auszugehen ist und somit sehr sensible Mikrofone ohne Bedenken eingesetzt werden können, werden Mikrofone für Field-Recordings zum Teil eher danach ausgewählt, das sie nicht aufzeichnen (Marks, 2017). Deshalb kommen im Field-Recording häufig sehr stark gerichtete Mikrofone, zum Beispiel *Richtrohre*, englisch auch *Shot-Gun-Mics* genannt, zur Anwendung. Diese haben dank ihrer *Keulencharakteristik* einen sehr schmalen Aufnahmebereich und blenden dementsprechend Geräusche außerhalb dieses Bereichs aus. Eine weitere Möglichkeit, die bei Ornithologen beliebt ist und auch zum Abhören in Detektivfilmen auftaucht, sind *Parabolspiegel-Mikrofone*. Der Gedanke dahinter ist, dass der Schall aus dem gewünschten Aufnahmebereich durch den Parabolspiegel an einem

²⁴ Druckempfänger sind tendenziell unempfindlicher gegenüber Körper-/Trittschall und Wind (Dickreiter, 2014).

²⁵ Kondensatormikrofone benötigen in der Regel eine 48V Phantomspannung, um das Elektrische Feld aufrecht zu erhalten (Dickreiter, 2014).

Punkt, an dem das Mikrofon sitzt, fokussiert wird. Im Unterschied zum Richtrohr, bei dem Schall aus seitlicher Einfallrichtung ausgelöscht wird, wird durch die Fokussierung des Schalls an einem Punkt die Amplitude des Nutzsignals verstärkt. In beiden Fällen wird so das Nutzsignal von Stör-signalen freigestellt. Durch die Amplitudenverstärkung liefern Parabolspiegel-Mikrofone jedoch einen höheren Pegel und benötigen vergleichsweise weniger Verstärkung. Dies wiederum führt zu weniger verstärkerbedingtem Rauschen (Hinrichsen, 2015).

Spezielle Anforderungen an Field-Recorder können beispielsweise eine einfache Handhabung sein, da der Sound Designer schnell und zuverlässig auf plötzliche Sounds reagieren können muss. Wenn Mikrofone Phantomspannung benötigen, muss der Recorder diese liefern können. Weitere Kriterien sind eine entsprechende Anzahl an Kanälen, Anschlüssen, Aufnahmequalität, Mikrofonvorverstärkung und Stromversorgung, beziehungsweise Akku- oder Batterielaufzeit (Marks, 2017).

Weiteres benötigtes Equipment kann von Batterien, über Speichermedien bis hin zu Ersatzgeräten reichen und sollte ebenfalls miteingeplant werden.

Insbesondere für Survival-Games die im Freien stattfinden und, wie zum Beispiel THE FOREST (2014) oder STRANDED DEEP (2015), das Fällen von Bäumen und Fördern anderer natürlicher Ressourcen beinhalten, kann eine Field-Recording-Session in einem Wald sehr von Nutzen sein.

4.2.5 Nachbearbeitung

Ziel dieses Arbeitsschrittes ist es, die Sounds fertigzustellen. Außerdem soll dabei die Klangqualität und der Klang der Sounds zusätzlich verbessert, sowie Störgeräusche und -signale entfernt werden. Die Sounds sollen insgesamt den Anforderungen des jeweiligen Funktionsfeld gerecht werden. Dabei können Edit und Pre-Mix parallel zueinander und im Wechsel auftreten und müssen nicht zwingend separat durchgeführt werden, was aus der Sicht eines Toningenieurs der Musik- oder Filmbranche seltsam anmuten mag.

In diesem Kapitel sollen allgemeine Aufgaben des Edits und Pre-Mixes, sowie soundgruppenspezifische Arbeitsschritte der Loop-Erstellung betrachtet werden. Es werden auch Vorteile einer objektorientierten Arbeitsweise, speziell im Edit und Pre-Mix von Computerspielton besprochen.

4.2.5.1 Entfernen von Störgeräuschen

Sowohl in Studioaufnahmen, als auch in Field-Recordings können allerhand Störgeräusche auftreten. Es ist als erstes zu klären, welcher Natur das Störgeräusch ist, bevor es in Angriff genom-

men werden kann. Mögliche Störsignale lassen sich in die Kategorien Brummen, Rauschen, impulsive Störgeräusche, Clipping, Dropouts, sowie sporadische, beziehungsweise einzelne Störgeräusche (iZotope Inc., 2014) unterteilen.

Im besten Fall ist alternatives, störsignalfreies Audiomaterial vorhanden. Bereits bei der Aufnahme sollte etwa darauf geachtet werden, dass das Signal nicht die 0 dB FS überschreitet, also kein Clipping auftritt und es auch nicht zu Dropouts kommt. Für den Fall, dass direkt eine Wiederholung dieses Sounds möglich ist, sollte dies auch so gelöst werden. Aus diesem Grund wird an dieser Stelle davon ausgegangen, dass das vorliegende Material frei von Clipping und Dropouts ist.

Gegen ein Brummen, wie etwa einem 50 Hz Netzbrummen, kann, insofern kein Nutzsignal in diesem Frequenzbereich vorliegt, ein tief angesetzter steiler Hochpass Abhilfe schaffen. Ansonsten ist ein Brumm-Filter die bessere Wahl; in der digitalen Umgebung treten sie häufig in Form eines De-Hum Plug-Ins auf (vgl. Abbildung 6). Bei einem Brummfilter wird die Grundfrequenz durch einen Notchfilter abgedämpft. Zusätzlich sind weitere Notchfilter für gerade und ungerade Harmonische zuschaltbar, sowie eine Abstufung der Dämpfung möglich (Slope-Funktion rechts unten in Abbildung 6) (Dickreiter, 2014).

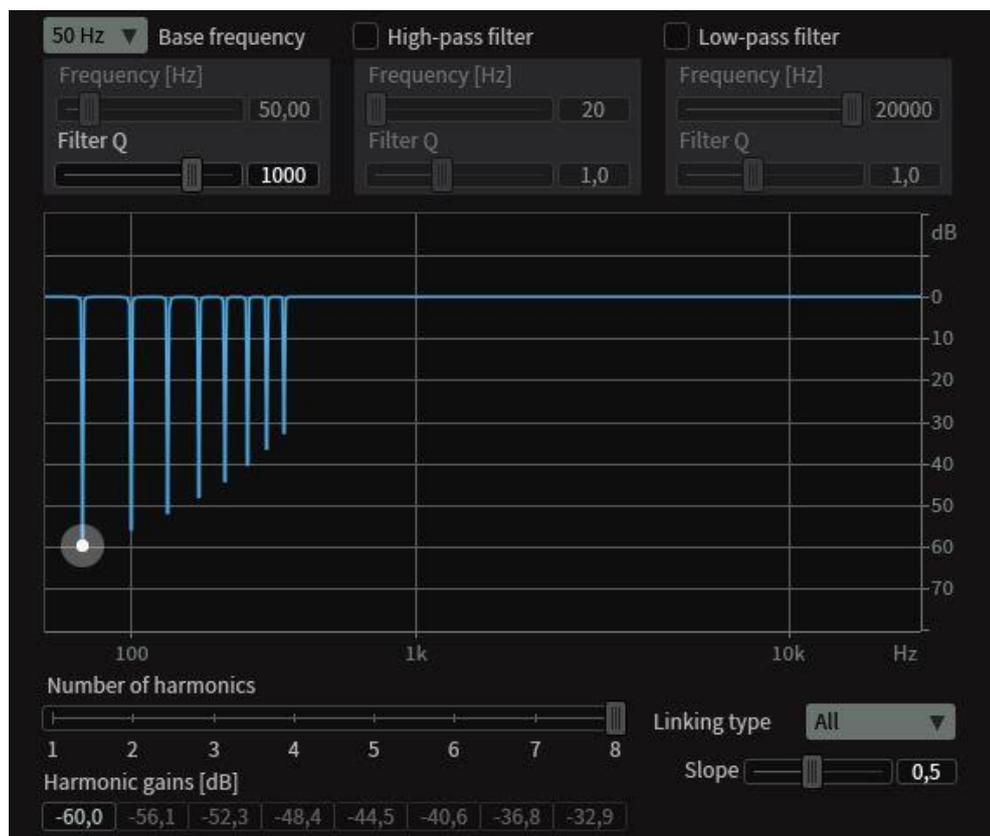


Abbildung 6: De-Hum Modul aus iZotope RX 6

Rauschen ist nach Dickreiter (2014, S. 3) als „Schallsignal statistischer Natur, bei dem nur ein kontinuierliches Frequenzspektrum angegeben werden kann“ definiert. Auch wenn ein Rauschen nicht unbedingt einer Behandlung bedarf und es abzuwägen ist, ob es nicht in der Aufnahme bleiben soll, so ist der De-Noiser eines der wichtigsten Tools der Störgeräuschreduzierung im Sound Design für Computerspiele. Ein De-Noiser kann dann notwendig werden, wenn sehr leise Geräusche aufgenommen wurden, welche im Nachhinein stark im Pegel angehoben oder komprimiert werden. Hierdurch wird neben dem Nutzsignal auch das Rauschen mitgehoben. Außerdem kann sich das Rauschen durch ein Layering aufsummieren.

Zusätzlich kann in Computerspielen nicht auf eine mögliche Verdeckung durch andere Audio-komponenten vertraut werden, da nie sicher bekannt ist, wann welche Sounds zusammen spielen, zumal in den meisten Spielen zusätzlich einzelne Bestandteile der Vertonung in den Optionen deaktiviert werden können (Vachon, 2009).

Angenommen ein gut wahrnehmbares Rauschen sei Bestandteil der Fußschritte. Mit jedem Aufrufen der Fußschritte ist dieses Rauschen solange zu hören, bis der Sound fertig abgespielt ist. Es entsteht ein ähnliches Dilemma, wie beim Einsatz eines Noise-Gates: das Rauschen ist zwar zwischen den einzelnen Schritten nicht zu hören, dafür während den Schritten in voller Lautstärke.

Der De-Noiser muss zunächst das Rauschen erkennen. Hierzu wird ein Noise-Sample benötigt, weshalb bei den Aufnahmen darauf geachtet werden sollte, auch etwas ‚Stille‘ zwischen den Nutzsignalen aufzunehmen. In iZotope’s RX (vgl. Abbildung 7) wird die Wellenform in blau und ein Spektrogramm (die Vertikale beschreibt die Frequenz, die Horizontale den Zeitverlauf und die Farbintensität entspricht der Amplitude) dargestellt.

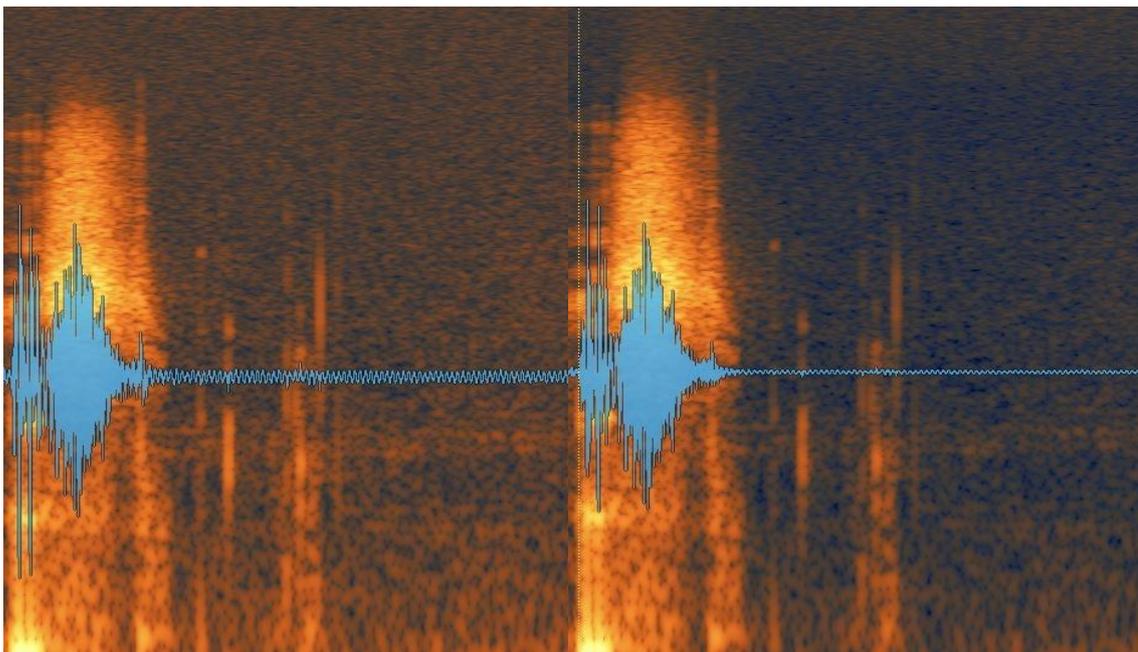


Abbildung 7: Spektrogramm und Wellenform aus iZotope RX 6

Das Rauschen ist hier als orange Flecken zu erkennen. Der De-Noiser rechnet anschließend das erlernte Rauschen aus dem Signal heraus (Abbildung 7, rechts). Es muss dabei auf mögliche hörbare „Artefakte in Form von Veränderungen der Klangstruktur und Klingeln, auch engl. Musical noise“ geachtet werden (Dickreiter, 2014, S. 393).

Zu den impulsiven Geräuschen gehören auch Klick-Geräusche. Diese können unterschiedlichen Ursprungs sein. Mundgeräusche, Schallplattenknistern aber auch Schnitte die nicht an Nulldurchgängen durchgeführt und nicht gefadet wurden, können Klicks verursachen. Sie sind im Spektrogramm als vertikale Linien zu erkennen. Impulsive Störgeräusche zeichnen sich allgemein dadurch aus, von nur kurzer Dauer zu sein (iZotope Inc., 2014).

Die einfachste, aber auch in ihren Möglichkeiten beschränkteste Methode, Audiomaterial von Klick-Geräuschen zu befreien, ist ein Herausschneiden dieser. Ein De-Clicker kann automatisiert helfen. Es ist bei der Verwendung eines De-Clickers notwendig darauf zu achten, dass kein Nutzsignal fälschlicherweise als Klick erkannt und gedämpft wird.

Hat auch dies den Klick nicht entfernt, ist eine Spektralbearbeitung, zum Beispiel mittels des *Spectral Cleanings* in MAGIX Sequoia, eine weitere Möglichkeit Abhilfe zu schaffen. Hierfür werden die Klicks in einem Spektrogramm markiert (vgl. Abbildung 8). Es kann nun eingestellt werden, in welcher Art der ausgewählte Bereich zu bearbeiten ist (überblenden, dämpfen, etc.), in welcher Auflösung und in welcher Intensität.

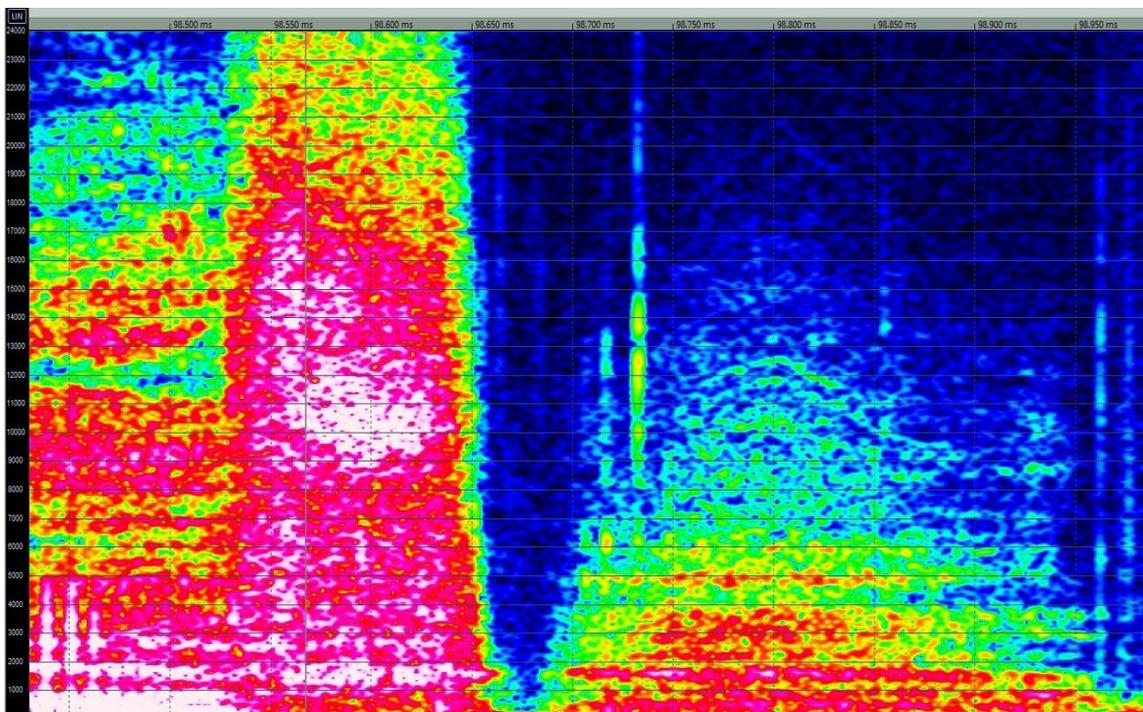


Abbildung 8: Spektrogramm aus Sequoia 14 mit deutlich erkennbarem Klick

Im Fall von einzelnen diversen Störgeräuschen wie etwa Husten oder etwa einem Autohupen im Hintergrund, kann eine erneute Aufnahme, falls dies denn möglich ist, die einfachste Lösung sein. Es kann funktionieren, Störgeräusche einfach herauszuschneiden und die Schnitte im Anschluss durch Crossfades zu kaschieren. Hierbei ist darauf zu achten, dass keine Sprünge, Pegel einbrüche oder Pegelzunahmen entstehen. Als Faustregel gilt: bei Schnitten in kohärentem Audiomaterial wird eine Linearblende verwendet, bei Schnitten in nicht-kohärentem Material eine logarithmische (Thornton, 2007). Ist ein Schneiden nicht möglich, so kann auch hier eine Spektralbearbeitung helfen.

Nachdem das Rohmaterial ‚gesäubert‘ wurde, kann es nun entweder direkt verwendet, oder aber noch nach Belieben verfremdet werden. Eine wichtige Methode ist dabei das Layering.

4.2.5.2 Layering (statisch)

Die in diesem Kapitel besprochene Art des *Layerings* wird in dieser Arbeit als *statisch* bezeichnet, um es vom *Echtzeit-Layering* in der Middleware abzugrenzen. Zwar basieren beide auf derselben Idee, jedoch verfolgt das Echtzeit-Layering einen erweiterten Funktionsumfang. Dies wird im Kapitel 5.1.8 genauer betrachtet.

Als *Layering* wird die Technik bezeichnet, einen neuen Sound aus verschiedenen Einzelkomponenten zu schaffen. Der Name bezieht sich auf die Anordnung dieser Einzelteile innerhalb einer DAW. Diese einzelnen Schichten werden mehr oder weniger gleichzeitig abgespielt; sie liegen vertikal untereinander (vgl. Abbildung 9).

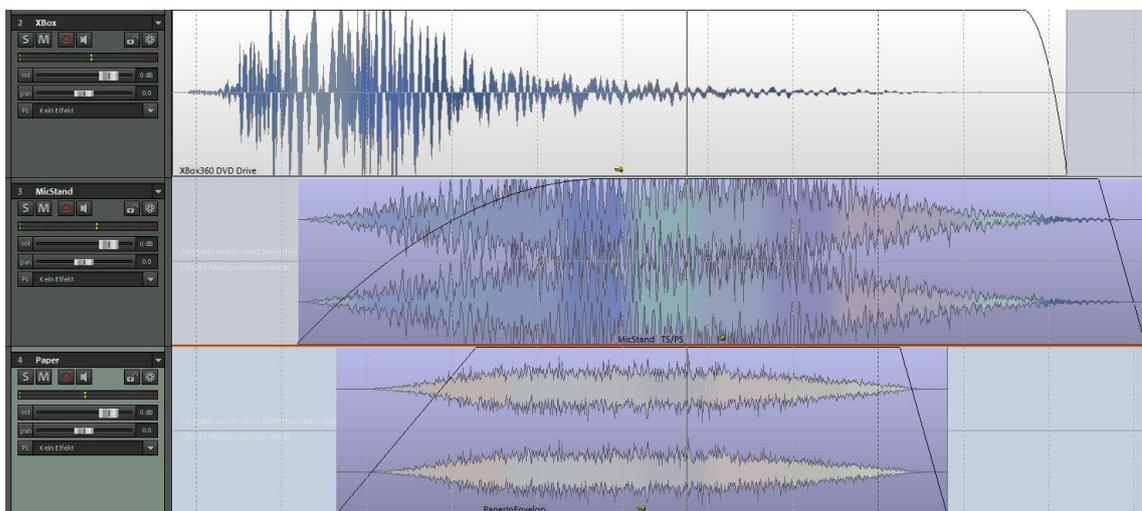


Abbildung 9: statisches Layering

Erfahrungsgemäß ist es beim Kreieren der Sounds hilfreich, Geräusche losgelöst von ihrem realen Ursprung und stattdessen nach ihren auditiven Bestandteilen zu analysieren. Da es bei der Aufnahme von Sounds häufig nicht praktikabel ist, einen echten Vertreter eines Objekts zu vertonen,

oder aber, wie im Fall von Sci-Fi oder Fantasy schlichtweg kein realweltliches Äquivalent existiert, kann es sich lohnen, einen auditiven Bauplan zu schaffen. Ziel dieses Bauplans ist es, einen Sound in seine Funktionsgruppen zu unterteilen.

Diese Funktionsgruppen können, beispielsweise Frequenzkomponenten betreffen, oder genauer definierte auditive Merkmale sein. Ein paar Beispiele:

- Das Geräusch des einfrierenden Levels in THE DARK SIDE OF THE MOON wurde in zwei Komponenten unterteilt: Das Knacken von Eis als hohe Komponente, die Definition mit einbringt und ein tiefes Gletschergrollen, um dem Gesamtklang mehr Gewicht und Druck zu verleihen. Gemischt wurde es anschließend recht tiefenbetont, um es insgesamt bedrohlicher und größer wirken zu lassen.
- Der Sound beim Einsammeln von Items sollte universell wirken, also nicht spezifische Objekttypen näher charakterisieren. Stattdessen sollte vermittelt werden, dass etwas aufgehoben und eingesteckt wird. Hierzu wurden diverse Geräusche des Anfassens verschiedener Gegenstände mit dem Einstecken in einen Rucksack und einem zufallenden Buch kombiniert.
- Die Türen der Raumstation sollten auditiv an die klassischen Sci-Fi Türen angelehnt sein, wie sie aus STAR WARS und STAR TRECK bekannt sind. Das gewünschte Klangobjekt kann in eine Art schleifendes, höertöniges Geräusch, einer hydraulisch anmutenden Komponente und außerdem einen mechanischen Verschluss unterteilt werden. Zusammengesetzt wurde der Sound aus dem Geräusch des Aus- und Einfahrens eines Mikrofonständers, sowie dem Geräusch eines Briefes der aus einem Briefumschlag gezogen wird und dem Öffnen des DVD-Laufwerks einer Xbox 360.

Die Arbeit des Sound Designers kann zum Teil stärker auf Assoziationen beruhen, als auf der Vertonung der Realität. Dies erfordert einiges an Übung und Kreativität. Es bestimmt gewissermaßen den akustischen Fingerabdruck eines Tonschaffenden.

Es ist auch sinnvoll, sich in diesem Zusammenhang die Frage zu stellen: Welche Klangobjekte können potentiell besonders unter einer Abnutzung leiden und ist es sinnvoll einzelne Layer für diese vorzubereiten um sie später etwa in Echtzeit zu kombinieren? Diese können wie ein statisches Layering kreiert werden, mit dem Unterschied, dass die Spuren separat, statt in der Summe gerendert werden.

Zusätzlich sollte darauf geachtet werden, dass die verschiedenen Klänge einer Art ähnlich genug klingen, damit sie nicht die Immersion gefährden; zum Beispiel wurde in THE DARK SIDE OF THE MOON darauf geachtet, dass die Geräusche der Schiebetüren insgesamt ähnlich klingen und keine einzelnen markant herausstechen. Werden Sounds gelayert, so können nicht nur der Gesamtklang,

sondern auch die einzelnen Komponenten entsprechend angepasst werden. Allgemein können die einzelnen Schichten separat bearbeitet und Verfremdet werden, beispielsweise mit Entzerrer, Kompressor und Pitch-Shifter.

4.2.5.3 Erstellen von Loops

Eine wichtige Fähigkeit, die jeder Sound Designer im Computerspielbereich benötigt, ist die Erstellung von einwandfreien *Loops*. Ein Loop ist ein Sound, der in Dauerschleife abgespielt, also geloopt wird.

Loops können gewissermaßen als Reaktion auf die Unklarheit des zeitlichen Spielablaufs gesehen werden. Durch Loops kann sich der Spieler für eine beliebig lange Zeit in einem Gebiet aufhalten; die Länge der Audiodatei ist losgelöst von der Aufenthaltsdauer des Spielers. Besonders bei Atmos und Musik sind Loops beliebt, wenn dies auch nicht die einzigen Einsatzgebiete sind (Leenders, 2012).

Um zu klären, was einen guten Loop ausmacht, ist es am einfachsten, das Worst-Case-Szenario zu betrachten und daraus Rückschlüsse zu ziehen. Dabei ist es gleich ob der Loop Musik, Atmo oder andere Geräusche beinhaltet. Im schlimmsten Fall ist ein Loop von sehr kurzer Dauer und wird sehr häufig wiederholt. Dabei sind zusätzlich sehr Markante und wiedererkennbare Bestandteile zu hören. Bevor der Loop von vorne beginnt, ist eine kurze Unterbrechung oder ein Sprung zu hören (Phillips, 2014).

Wichtig ist, dass ein Loop nach Möglichkeit nicht erkannt wird und so eine Abnutzung vermieden wird. Je länger ein Loop ist, desto geringer ist die Zahl der benötigten Wiederholungen innerhalb einer Situation; jedoch wird dadurch auch der Speicherbedarf erhöht (Leenders, 2012). Das Material sollte möglichst frei von wiedererkennbaren Elementen und Mustern sein, da der Spieler anhand dieser sehr schnell die Wiederholung entlarven kann. Zur Verdeutlichung ein Beispiel von O'Donnell (2002): „[...] the biggest tip off to the listener that something is artificial is when the crow always caws just after the leaf rustle and before the frog croak every thirty seconds [...].”

Beim Erstellen von Loops können Hallfahnen unter Umständen ein Problem darstellen, da ein abruptes Abbrechen des Halls, oder ein plötzliches Einsetzen unnatürlich klingt (Phillips, 2014).

Um einen makellosen Loop zu erstellen, muss er mindestens einigermaßen so beginnen wie er geendet ist (Phillips, 2014). Eine Möglichkeit ist es durch Trial und Error Stellen im Audiomaterial zu finden, welche sauber ineinander übergehen.

Doch es gibt auch eine zuverlässigere Methode. In *THE DARK SIDE OF THE MOON* wurde beispielsweise für das Knistern des Feuers ein Loop erstellt. Dabei gliedert sich die Erstellung des Loops in vier Schritte:

1) Anlegen des Sounds

Zunächst wird der zu loopende Sound in der DAW platziert und, falls dies noch nicht geschehen ist, von sämtlichen Störgeräuschen und wiedererkennbaren Stellen bereinigt.

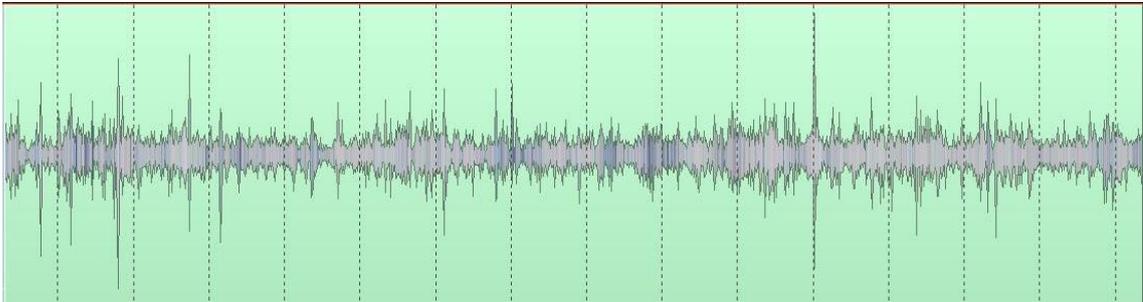


Abbildung 10: Loop-Erstellung, das Ausgangsmaterial

2) Schnitt an einem Nulldurchgang

Als nächstes wird ein Nulldurchgang im Audiomaterial gesucht und der Audio-Clip aufgeteilt. Durch den Schnitt im Nulldurchgang werden Knacken und Sprünge vermieden.



Abbildung 11: Loop-Erstellung, Schnitt an einem Nulldurchgang

3) Tausch der beiden Audio-Clips

Die beiden Clips werden in ihrer horizontalen, also zeitlichen Anordnung getauscht. Sprich der vordere Clip liegt nun hinten und der zuvor hintere liegt nun vorne. Hierdurch ist gewährleistet, dass der Loop so beginnt wie er geendet ist.

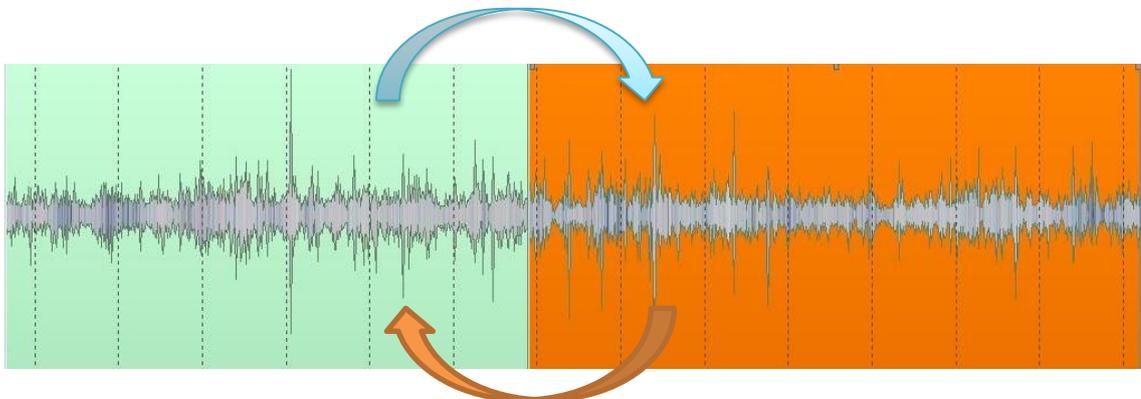


Abbildung 12: Loop-Erstellung, Tauschen der Clips/Objekte

4) Unauffälliges Überblenden

Um nun keinen Sprung mitten im Loop zu erzeugen, wird ein Crossfade eingesetzt. Da hier im selben Material übergeblendet wird, ist eine logarithmische Blende gewählt worden. Gegebenenfalls ist ein Experimentieren mit der genauen Anordnung vonnöten, damit der Schnitt auch wirklich nicht hörbar ist.

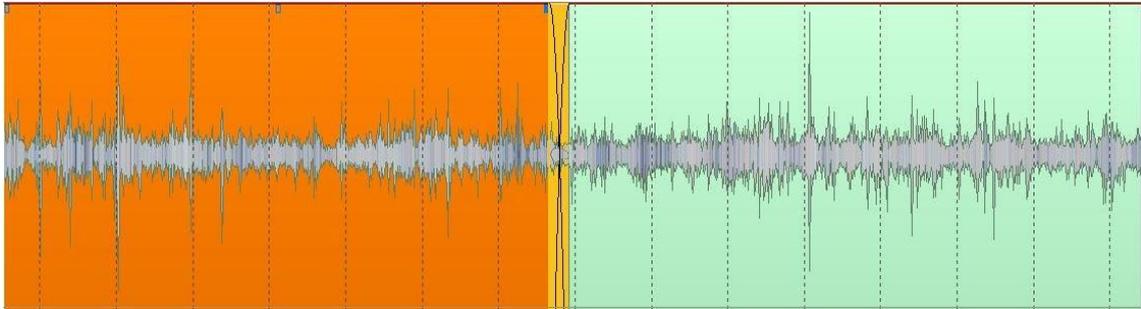


Abbildung 13: Loop-Erstellung, Crossfade anlegen

Das Ergebnis ist ein sauberer Loop, frei von Sprüngen und Knackern. Dieses Vorgehen ist für allerlei Geräusche geeignet. Für Musik bietet sich diese Methode logischerweise nur bedingt an, da ein willkürlicher Schnitt potentiell auch an musikalisch ungünstigen und kritischen Stellen sitzen kann und eventuell das Stück sogar unbrauchbar macht. Auf diese Weise kann jedoch beispielsweise dafür gesorgt werden, dass der Nachhall eines Stückes in den Anfang übergehen, ohne ihn offensichtlich zu beschneiden oder komplett ausklingen zu lassen.

4.2.5.4 Pre-Mix

Ziel des Pre-Mixes ist es, die einzelnen Sounds einer Art, als auch im Gesamten untereinander anzupassen, so dass ein insgesamt einigermaßen einheitliches Klangbild entsteht. Auch wenn nie genau bekannt ist, wann was zu hören ist, sollte bereits die Überlegung angestellt werden, welche Sounds sehr wahrscheinlich gleichzeitig im Spiel zu hören sein werden und dementsprechend der Frequenzbereich aufgeräumt werden. Es ist zu bedenken, dass sich einzelne Sound-Ebenen in der Regel in den Optionen ausschalten lassen. Das Klangbild sollte auch in dieser reduzierten Version des Spiels gut klingen (Leenders, 2012).

Ebenfalls empfiehlt es sich, Pegel bereits grob anzupassen und beispielsweise sehr leise Geräusche auf eine brauchbare Lautstärke zu bringen (Horowitz & Looney, 2014). Der Vorteil dies bereits in der DAW zu erledigen besteht darin, dass bisher möglicherweise unentdeckte Störgeräusche und Fehler bereits frühzeitig entdeckt und behoben werden können. Fällt dies erst in der Middleware auf, muss gegebenenfalls die richtige Session und darin der Sound gesucht, neu bearbeitet, im Klangbild angepasst und exportiert werden.

Man könnte den Pre-Mix auch als eine Art Mastering bezeichnen; es nimmt eine ähnliche Rolle wie die des Masterings einer Musikproduktion ein, wenn auch nicht mit einer vergleichbaren Präzision. Der letzte Feinschliff findet, ebenso wie die endgültige Balance der Sounds davor in der Middleware, in Echtzeit statt. Der Pre-Mix ist also eher als Vorbereitung für die Middleware zu verstehen.

Es gibt grob gesagt zwei Möglichkeiten, diesen Pre-Mix umzusetzen. Entweder werden die Sounds in ihrer jeweiligen Bearbeitungs-Session (also der Session in der beispielsweise gelayert wurde) optimiert und blind gemischt. Oder aber, was eher zu empfehlen ist, die einzelnen Sounds werden gebounct und in eine eigene Mix-Session geladen und hier im Kontext zueinander bearbeitet (Material das später in Echtzeit gelayert werden soll, muss logischerweise in seinen einzelnen Layern gebounct werden).

Hierbei kommt es zu einer erheblichen Menge einzelner Sounds. Um Nuancen der Sounds einer Art aneinander anzupassen, wird eine ebenfalls große Zahl an Automationen benötigt, da eine statische Effekteinstellung in der Regel nicht zur Optimierung diverser Exemplare taugt. Durch statische Einstellungen lässt sich in der Regel eher eine bereits einheitlich klingende Gruppe von Sounds an eine andere anpassen.

Objektbasiertes Arbeiten kann hier eine deutliche Zeitersparnis mit sich bringen. Beispiele für objektbasiert arbeitende DAWs sind Samplitude und Sequoia von MAGIX. Hier liegen die einzelnen Sound-Clips als eigene Objekte vor. Jedes Objekt verfügt über einen eigenen, spurunabhängigen Kanalzug. Dies bedeutet, für jedes Objekt können Pegel, Panorama, Phase und nahezu sämtliche Plug-Ins separat eingesetzt werden. Anstatt Automationen zu fahren, kann das Objekt also einfach mit einem neuen Effekt geformt werden. In der Abbildung 14 ist beispielsweise das orange Objekt ausgewählt. Unten ist der Kanalzug zu sehen. Der Pegel des Objekts wurde um circa 12 dB angehoben. Zusätzlich liegen ein EQ und ein Kompressor auf dem Objekt, eigens für dieses eingestellt.

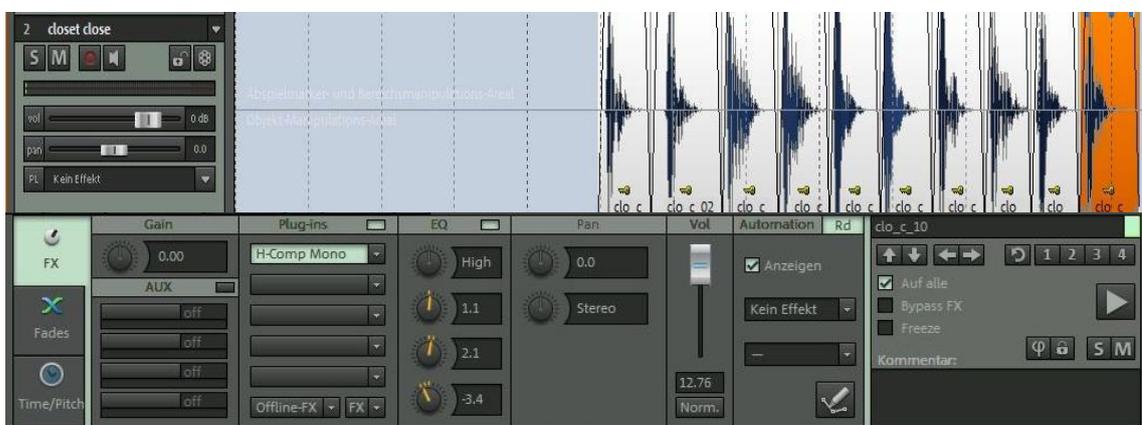


Abbildung 14: Objektbasiertes Arbeiten im Pre-Mix

Zusätzlich zur klanglichen Anpassung ist es wichtig, dass die Sounds entsprechend ihrer Aufgabe gleich eng beschnitten werden, also das Signal nach einer möglichst gleichen Zeit einsetzt. Ist Stille vor dem eigentlichen Signal, so wird auch erst diese abgespielt. Im Fall von einzeln getriggerten Fußschritten kann ein ungleichmäßiger Einsatz dazu führen, dass der Spielcharakter zu humpeln scheint oder als ob er unter anderen Gehproblemen leidet.

Zusätzlich kann durch das enge Beschneiden immer auch ein Quäntchen Speicherplatz eingespart werden. Darüber hinaus ist darauf zu achten, dass Sounds sauber starten und enden, ohne Klicks, Pops, oder sonstigen Fehlern (Horowitz & Looney, 2014).

4.3 Exkurs – Voice Over und Voice Acting

Wenige Bereiche einer Computerspielvertonung können den Eindruck eines Spiels so schnell zerstören, wie ein schlechtes Voice Acting und Voice Over. Dabei sind die Stolpersteine überall verteilt. Schlechte Qualität des Skripts oder der Spracheaufnahmen selbst, mangelnde Schauspielkunst, fehlerhaftes Editing oder eine schlechte Implementierung, all diese Gefahren gilt es zu vermeiden. Die Stimme kann in Computerspielen das einzige wirklich echte menschliche Element sein, weshalb Missstände in diesem Bereich schnell auffallen (Marks, 2017).

Bedingt durch den, im Vergleich zum Film anderen Produktionsablauf, ist es nicht unüblich, dass ein Voice Acting ohne festen Bildbezug, sondern auf Konzepten und Storyboards erstellt werden muss. Dies kann dem Voice Actor das Finden des richtigen Tons des jeweiligen Charakters erschweren (Leenders, 2012). Zusätzlich kommt noch hinzu, dass oft kein zusammenhängendes Skript vorliegt, sondern viele einzelne, zum Teil voneinander losgelöste Phrasen aufgenommen werden (Horowitz & Looney, 2014). Häufig wird nicht das Voice Acting auf die Lippen der Charaktere vorgenommen, sondern andersherum, die Lippenbewegungen anhand des aufgenommenen Materials nachsynchronisiert (Marks, 2017). Es ist also darauf zu achten, dass die Performance den jeweiligen Charakter entsprechend repräsentiert und die Intensität durchgehend konstant, beziehungsweise angemessen bleibt.

Funktionen von Sprache in Computerspielen können sein: Feedback über den aktuellen Stand, wie etwa Hinweise was als nächstes zu tun ist; ein Erzähler, der Hintergrundinformationen oder Anweisungen gibt; Dialoge und Monologe (Horowitz & Looney, 2014).

Es ist abzuklären, in welcher Form die Sprache vorliegen soll. Häufig wird das Voice Over zeilenweise getriggert. Insgesamt können Computerspiele mit Filmen in der Menge des Gesprochenen mithalten oder sie sogar überholen. Halo hatte beispielsweise mehr als sechstausend Zeilen Dialog (O'Donnell, 2002).

Da sich die eigentliche Produktion von Sprachaufnahmen nicht allzu sehr von der Produktion für andere Medien unterscheidet und der Fokus dieser Arbeit insbesondere auf dem Sound Design liegt, wird an dieser Stelle nicht weiter hierauf eingegangen. Zumal Survival-Games eher zu den dialogarmen Spielen gehören; THE FOREST (2014), STRANDED DEEP (2015) und DAYZ (2012) kommen sogar bisweilen gänzlich ohne aus.

4.4 Exkurs – Musik

Die Musik für Computerspiele kann völlig unterschiedlich umgesetzt werden. Wird im Kontext von Computerspielen von linearer Musik gesprochen, so kann damit verschiedenes gemeint sein. In linear ablaufenden Spielen wie zum Beispiel dem Shooter BATTLEFIELD: BAD COMPANY 2 (2010) können für jeden Abschnitt jeweils eine eigene geloopte Musik angelegt sein. Oder aber es liegen abgeschlossene, linear ablaufende Musikstücke vor, die an die frei begehbaren Orte gebunden sind. Ein Beispiel hierfür kann in THE WITCHER 3: WILD HUNT (2015) gefunden werden. Die verschiedenen Distrikte der Stadt Novigrad haben jeweils ihr eigenes Stück zugeordnet. Die Musik kann dabei immer noch dynamischer sein. Die Linearität ist in diesem Fall eher auf den Ablauf der Musik selbst zurückzuführen (Phillips, 2014; Rumsey, 2010). Allgemein ist die Verwendung von linearen Stücken in RPGs etwa einfacher umzusetzen, da es durchaus genrekonform sein kann, nach dem Ende eines Musikstücks eine Pause zu lassen, in der keinerlei Musik spielt (Leenders, 2012).

Da es nicht möglich ist, wie im Film, einen langen Track für das gesamte Spiel zu haben, muss die Musik wiederholt werden. Nicht in jedem Spiel und /oder Genre sind solche erwähnten Pausen praktikabel oder angebracht. Falls sich ein und dasselbe Stück direkt wiederholen soll, muss es so komponiert sein, dass es nach Möglichkeit nahtlos in sich selbst übergehen kann.

Um den Bedürfnissen eines Computerspiels gerecht zu werden, also dennoch Abwechslung zu schaffen, den Score länger wirken zu lassen und die Musik dynamisch zu gestalten, herrschen vor allem zwei Methoden vor:

1) Horizontale Re-Sequenzierung:

Bei der horizontalen Re-Sequenzierung besteht der Score aus unterschiedlichen Stücken, deren Abfolge an Parameter geknüpft ist und in Echtzeit neu arrangiert wird. Es muss folglich beim Komponieren stets einiges im Auge behalte werden, damit dieses Neuarrangieren auch glatt und unauffällig funktioniert, wie etwa die Tonart, die Harmonien, das Tempo und der Rhythmus (Phillips, 2014). Ein Beispiel für diese Art von Musik wäre der Wechsel von einer friedlichen Musik beim Erkunden der Welt, hin zu einer Kampf-musik sobald sich Feinde in der Nähe befinden und wieder zurück zur friedlichen, sobald

alle Feinde besiegt sind. Dieses wird so beispielsweise in THE WITCHER 3: WILD HUNT (2015) als auch in THE ELDER SCROLLS V: SKYRIM (2011) gelöst.

2) Vertikales Layering:

Genau wie im Sound Design besteht die Musik aus unterschiedlichen Schichten; häufig werden hierzu verschiedene Instrumente zu Stems zusammengefasst. Die Komposition kann durch das Hinzuschalten oder Ausblenden dieser Schichten unterschiedliche Gestalten annehmen, auch wenn im Kern immer dasselbe Stück gespielt wird (Phillips, 2014). Ein Beispiel hierfür ist das Spiel RED DEAD REDEMPTION (2010). Hier wurde die gesamte Musik in A-Moll und mit 130 BPM aufgenommen um sicherzugehen, dass verschiedene Schichten problemlos ausgetauscht werden können. Sitzt der Spieler etwa auf ein Pferd auf, wird beispielsweise eine Bass-Line eingeblendet (GamerSpawn, 2010).

Die Erwartungen seitens der Spieler an die Musik und ihr Einsatz sind immer auch an das jeweilige Genre gebunden. Survival-Games kommen zum Teil sogar mit beinahe keiner Musik aus, wie beispielsweise THE FOREST (2012). Musik spielt hier nur in zwei Fällen: zum einen, wenn der Spieler von Feinden erblickt wurde, so ertönt ein kurzes Pianomotiv (auch wenn dies eher die Funktion eines Sound FX einnimmt); zum anderen wenn der Spieler Musikkassetten und Walkman findet und benutzt.

In THE DARK SIDE OF THE MOON wurde ein vertikales Layering mit einer Technik verwendet, die Phillips als „Slow Textures“ bezeichnet (Phillips, 2014, S. 166). Hierbei besteht die Musik aus lange stehenden Akkorden, die langsam ineinander übergehen. Der Übergang dieser Art von Musik zu Atmos hin ist recht fließend.

5 Middleware

Um den Nutzen einer Middleware besser zu verstehen, ist es hilfreich, zunächst den Workflow von früher zu betrachten. Dieser sah so aus, dass der Sound Designer die fertigen Sounds an den Programmierer übergab, der diese dann implementierte. Erst nach dieser zeitintensiven Implementation konnte der Sound Designer seine Arbeit im Spielkontext evaluieren. Sollen Änderungen vorgenommen werden, bedeutete dies, dass jeder veränderte Sound auch neu implementiert werden musste. Daraus folgt, dass jede Änderung eines Sounds auch eine Änderung des Programmcodes bedeutete. Überspitzt gesagt, entscheidet der Programmierer über die finale ‚Mischung‘ und den Klang des Spiels, da er die Balance, aber auch das Triggerverhalten und etwaige Übergänge von Sounds (hart) codiert (Leenders, 2012). Dabei muss davon ausgegangen werden, dass der Programmierer nicht auch zusätzlich ein Sound Designer ist und ein nicht optimales Ergebnis dabei herauskommt, welches an der Vision des Sound Designers vorbeigeht. Von diesem Standpunkt aus betrachtet, wäre ein System sinnvoll, in welchem der Sound Designer definieren kann, welches Event welche Sounds auf welche Weise aufruft. Dieses System sollte zudem losgelöst von der Game Engine in Betrieb genommen werden können (Schmidt, 2003).

In einer Middleware kann ein komplexes Soundverhalten definiert werden. Dabei ist die Middleware ein Tool, welches mit einer GUI bedienbar ist und somit ein Arbeiten ohne Programmierkenntnisse erlaubt (auch wenn diese hilfreich sind). Zu diesem Soundverhalten zählt zum Beispiel das Triggerverhalten. Wichtig ist es, dass die Parameter zwischen Sound Designer und Programmierer geklärt sind, damit die finale Einbindung problemlos verläuft (Horowitz & Looney, 2014). Weitere Vorteile die eine Middleware mit sich bringt sind eine robuste Audio Engine, eine Kompatibilität mit diversen Audioformaten, sowie eine plattformübergreifende Ausrichtung (Marks, 2017).

Alles in allem gewinnt somit der Sound Designer mehr Kontrolle über den Klang des Computerspiels. Zusätzlich wird eine effektivere Aufgabenverteilung zwischen Sound Designer und Programmierer ermöglicht: Der Sound Designer implementiert, der Programmierer integriert (Horowitz & Looney, 2014). Bei aller daraus resultierender Zeitersparnis, muss dennoch bedacht werden, dass die Schaffung eines Computerspiels stets ein asynchroner Prozess ist. Auch trotz Middleware muss der Sound zunächst in das Spiel eingebunden werden, bevor er getestet werden kann. Eine Middleware kann diesen Prozess lediglich verkürzen, nicht jedoch überwinden (Leenders, 2012).

Es gibt eine Vielzahl unterschiedlicher Middlewares, von denen einige als interne Tools der Entwicklerstudios geschrieben wurden. Als ein Urvater der Middleware kann beispielsweise bereits das Programm iMUSE gesehen werden, welches für die dynamische Vertonung von MONKEY

ISLAND 2: LECHUCK'S REVENGE (1991) geschrieben wurde. Während im Laufe der Zeit zunächst plattformspezifische Middlewares hervorkamen, nimmt die Verwendung von plattformübergreifenden Middlewares immer mehr zu. Dies lässt sich aus der theoretischen Überlegung her begründen, dass die Beschränkung auf eine einzige Middleware, also einen Industriestandard, eine zeitintensive Einarbeitung und Kennenlernphase am Anfang eines Projekts obsolet macht (Leenders, 2012).

5.1 FMOD Studio

Im Folgenden sollen einige Module und Tools der Middleware FMOD Studio von Firelight Technologies, sowie eine mögliche Arbeitsweise mit dieser betrachtet werden. An dieser Stelle wird ein Verständnis über die grundlegende Navigation und die Benutzeroberfläche des Programms vorausgesetzt. Wichtig: wird FMOD verwendet, so muss dies auch in der jeweiligen Game Engine als Plug-In aktiviert werden, da sonst Fehlermeldungen auftreten.

5.1.1 Audio Bin

In FMOD gibt es zwei grundlegende Möglichkeiten, Assets in ein Event zu laden. Die einfachste Methode ist es, die gewünschte Datei von der Festplatte per Drag-and-Drop in eine Spur zu ziehen. Hierdurch wird diese Datei im Hintergrund in den Assets Ordner kopiert und eine Referenz im *Audio Bin* erstellt. Die Assets sind in diesem Fall nicht in einer Ordnerstruktur sortiert; diese müsste nachträglich manuell angelegt werden.

Die zweite Methode besteht darin, die Audio-Assets des Spiels zunächst alle in den *Audio Bin* zu laden²⁶ und die benötigten Assets aus dem Audio Bin in gewünschte Spur zu ziehen. Die Assets können bereits in einer Ordnerstruktur geladen werden, oder auch in neue Ordner unterteilt werden. Zusätzlich kann die Hierarchie schnell geändert und die Assets umsortiert werden.

Es ist wichtig zu verstehen, dass FMOD mit den erstellten Kopien im Assets Ordner arbeitet und nicht die Originaldateien zurate zieht. Dadurch, dass die Assets in den einzelnen Events lediglich referenziert sind können sie mehrfach verwendet werden, ohne dass der Speicherbedarf zunimmt (Schütze & FMODTV, 2012).

²⁶ Der Audio Bin kann durch das Dropdown-Menü Window → Audio Bin, oder die Tastenkombination Strg+3 erreicht werden.

Der Vorteil eines Audio Bins besteht darin, weniger fehleranfällig zu sein: Arbeiten mehrere Personen an einer FMOD Session, ist sichergestellt, dass stets alle Beteiligten die benötigten Assets besitzen. Im Spiel noch nicht verwendete Assets sind mit #unused markiert. Zusätzlich ist eine Suche der Assets in FMOD möglich.

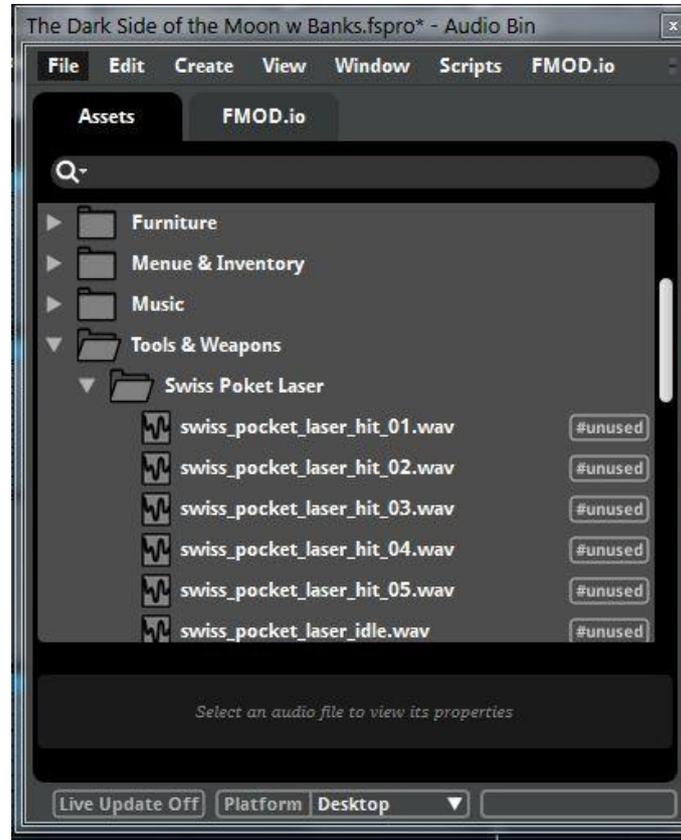


Abbildung 15: FMODs Audio Bin

5.1.2 Events

Events stellen in FMOD die grundlegenden Objekte dar, mit denen ein Sound kontrolliert und abgespielt wird. Diverse Module können innerhalb eines Events platziert und unterschiedliche Parameter definiert werden.

Ein neues Event kann durch das klicken auf New Event im Eventbrowser oder durch einen Rechtsklick in den Eventbrowser und anschließender Auswahl im erscheinenden Dropdown Menü erstellt werden.

Es gibt zwei unterschiedliche Arten von Events in FMOD, zwischen denen man beim Anlegen eines neuen Events entscheiden kann: 2D und 3D Events.

5.1.2.1 3D Event

Wird eine diegetische, sich in der Spielwelt lokalisierbare Quelle vertont, so wird ein 3D Event erstellt. Damit diese Quellen im Spiel geortet werden können, wird die Relation zum Listener genutzt, um das Resultierende Signal zu berechnen. 3D Events zeichnen sich dadurch aus, dass sich ihr Klang mit dieser Relation zum Listener verändert (Sound Librarian, kein Datum).

Um diese Beziehung zu verändern steht in FMOD zusätzlich ein *3D Panner* zur Verfügung. Hier kann unter anderem der Pegelabfall in Abhängigkeit der Entfernung variiert werden, die Ausdehnung der Schallquelle des Events, sowie die minimale und maximale Distanz in der das Event hörbar ist definiert werden (vgl. Abbildung 16).



Abbildung 16: 3D Panner

Wird dieser 3D Panner gelöscht, so wird die Position des Sounds von FMOD automatisch in ein 2D Stereofeld gesetzt. Dies bedeutet, wird der 3D Panner eines 3D Events gelöscht, so ist es nun ein 2D Event (Sound Librarian, kein Datum).

5.1.2.2 2D Event

Ein 2D Event wird standardmäßig in Stereo (Abbildung 17 links) oder Mono (rechts) angelegt; dementsprechend verfügt das Event in Stereo über einen einfachen Stereopanner; im Falle von Mono über gar keinen.

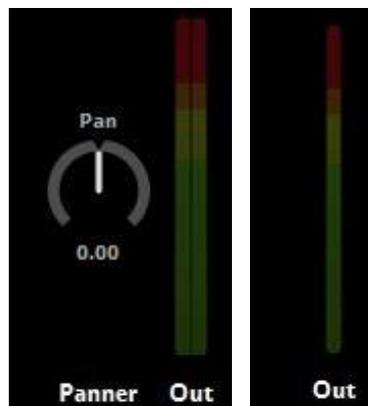


Abbildung 17: Stereo Panner links, Mono rechts (kein Panner)

Sie werden insbesondere für nicht lokalisierbare Sounds genutzt, wie etwa nichtdiegetische Musik, oder auch diegetische Klänge, wie zum Beispiel Atmos. Zusätzlich können Geräusche, die an der Position der eigenen Spielfigur geortet werden sollen, mittels eines 2D Events umgesetzt werden, wie etwa Fußschritte (insofern keine Unterscheidung zwischen Links und Rechts stattfinden soll) und auch die Stimme der Spielfigur selbst.

In manchen Spielen, wie etwa in *THE WITCHER 3: WILD HUNT* (2015)²⁷, wird auf eine solche Weise auch wichtiger Dialog gehandhabt, zum Beispiel als sich der Protagonist Geralt von Riva zusammen mit der Magierin Triss Merigold durch die Abwasserkanäle der Stadt Novigrad kämpfen: Durch die Verwendung eines 2D Event ist das Hörerlebnis unabhängig von der Entfernung zum sprechenden NPC, wodurch dieser Dialog auch in unübersichtlichen und turbulenten Spielsituationen gut hörbar bleibt. In diesem Fall fungieren 2D Events ähnlich dem Center Speaker einer Filmmischung im 5.1 oder 7.1 Format. Zwar wird somit sichergestellt, dass der Dialog präsent ist, jedoch wird dieser auch aus einer künstlichen Tonperspektive wahrgenommen. Es ist also abzuwägen, ob dies der Immersion nicht womöglich zu sehr schadet, insbesondere wenn Vocal Sound FX als 3D Event implementiert sind.

5.1.3 Soundmodule

Innerhalb der Events können unterschiedliche *Soundmodule* geladen werden. Diese Soundmodule können als die Art und Weise verstanden werden, wie die einzelnen Assets im jeweiligen Event aufgerufen werden.

Jedes Soundmodul wird durch eine *Trigger Region* in Form eines farbigen Kastens repräsentiert (vgl. Abbildung 18); entweder auf der Timeline oder in den Parametern selbst (Sound Librarian, kein Datum).

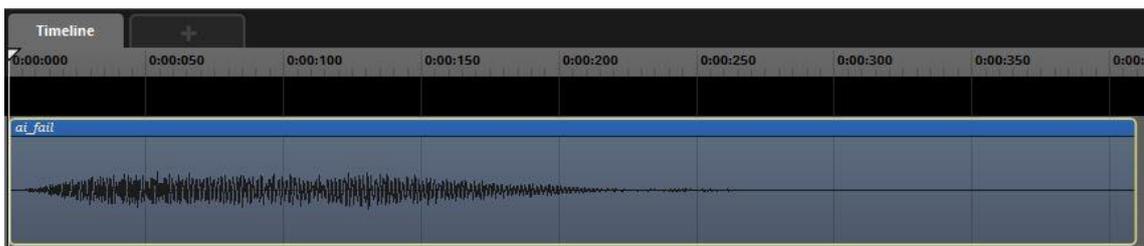


Abbildung 18: Trigger Region

Ein Event wird automatisch beendet, wenn es das Ende seiner letzten Trigger Region erreicht hat (sofern nicht geloopt wird). Es sollen im Folgenden die Module *Single Sound*, *Multi Sound*, *Scatterer Sound* und *Event Sound* vorgestellt werden.

²⁷ In den Credits von *THE WITCHER 3: WILD HUNT* (2015) wird darauf HINGEWIESEN, dass für die Vertonung des Spiels die Middleware Wwise eingesetzt wurde. Die Möglichkeiten dieser Middleware sind recht ähnlich zu denen, die FMOD bietet.

5.1.3.1 Single Sound

Ein *Single Sound* enthält, wie der Name bereits vermuten lässt, nur ein einziges Asset. Erkennen kann man dieses Modul an der einzelnen Wellenform, sowohl in der Visualisierung im Editor, als auch im Deck des Moduls. Single Sounds sind im Editor mit dem jeweiligen Asset-Namen beschriftet (vgl. Abbildung 19).

Single Sounds bieten sich für Sounds an, die stets gleich bleiben sollen, wie etwa UI-Sounds oder, wie im Fall der Abbildung, für Sounds die nur einmalig im Spiel auftreten und daher nicht unter einer Abnutzung leiden können; der Sound in der Abbildung wird nur beim ersten Erblicken der Erde ausgelöst.

Ein Single Sound wird einmalig abgespielt, sofern kein Loop aktiviert ist (ausgerautes Kreis-Pfeil-Icon über der Wellenform im Deck).

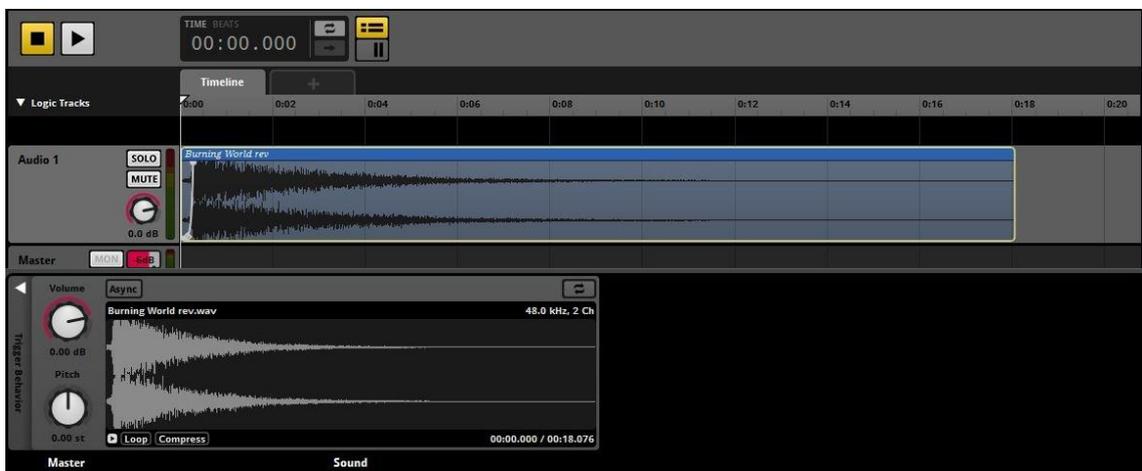


Abbildung 19: Single Sound

5.1.3.2 Multi Sound

Ein *Multi Sound* fungiert als Container für mehrere Assets. Erkennen kann man ihn an den Wellenformen, sowohl im Editor, als auch im Deck des Moduls. Im Editor ist ein Multi Sound auch als ein solcher beschriftet. Zusätzlich sind im Deck sämtliche Assets aufgelistet, welche Teil des jeweiligen Multi Sounds sind (vgl. Abbildung 20).

Multi Sounds bieten sich überall da an, wo eine Varianz benötigt wird. Mithilfe des kleinen Würfels über den aufgelisteten Assets kann eingestellt werden, ob die Assets in zufälliger Reihenfolge (Würfel = gelb) oder in der aufgeführten Reihenfolge (Würfel = grau) aufgerufen werden sollen. Auf weitere Möglichkeiten der Randomisierung wird später in einem eigenen

Kapitel eingegangen. Multi Sounds werden, genau wie Single Sounds nur einmal abgespielt, sofern sie nicht geloopt sind.

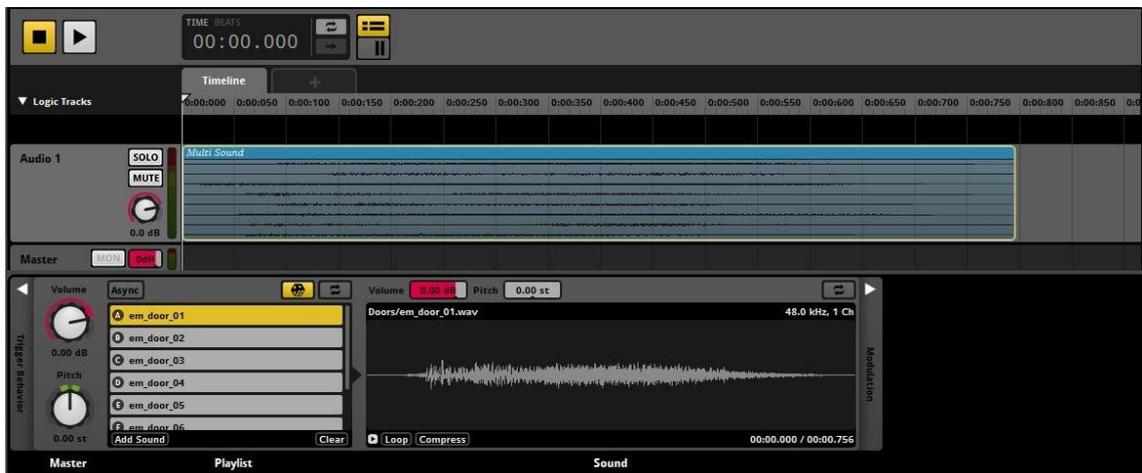


Abbildung 20: Multi Sound

5.1.3.3 Scatterer Sound

Ein *Scatterer Sounds* ist, ähnlich wie der Multi Sound, ein Containermodul, unterscheidet sich jedoch hinsichtlich seiner Einstellungsmöglichkeiten und letztlich seiner Komplexität.

Ein Scatterer Sound ist im Editor ebenfalls als solcher benannt; auch im Deck sind seine Bestandteile aufgeführt (vgl. Abbildung 21). Zusätzlich erlaubt ein Scatterer Sound, seine Bestandteile mehrfach abzuspielen und diese unterschiedlich im 3D Raum zu positionieren.

Die einstellbaren Funktionen sind (Sound Librarian, kein Datum):

a) Interval Between Sounds:

Hier kann das Triggerverhalten des Scatterer Sounds bestimmt werden, genauer welche *Zeit* mindestens und maximal zwischen dem Aufrufen zweier Assets liegen darf. Zusätzlich kann mit *Polyphony* definiert werden, wie viele maximal gleichzeitig ertönen können. Durch *Steal* kann bestimmt werden, was geschehen soll, wenn die maximale Zahl bereits erreicht ist und ein weiterer getriggert wird: Ist *Oldest* aktiviert, so wird das zeitlich am frühesten aktivierte Asset gestoppt und ein neues gestartet; bei *None* wird keins gestoppt und auch kein neues gestartet. Bei Erreichen der *Total Sounds* wird der Scatterer Sound vorzeitig beendet.

b) Min & Max Scatter Distance

Durch diesen Wert wird die räumliche Verteilung definiert, in welcher die Assets zufällig positioniert werden können. Es wird also der minimale und maximale Abstand zu dem, in der Game Engine definierten Emitter festgelegt.

c) Vol Rnd und Pitch Rnd

Hier kann eine Randomisierung (Rnd) für die Lautstärke (Vol) und die Tonhöhe (Pitch) des jeweils gespielten Assets definiert werden.



Abbildung 21: Scatterer Sound

5.1.3.4 Event Sound

Ein *Event Sound* stellt eine Referenz auf ein anderes Event dar; das Erstellen dieses Moduls generiert das dazugehörige Event automatisch. Sowohl der Event Sound, als auch das neue Event sind standardmäßig als *Nested Event* benannt. Wird das Event umbenannt, so ist auch das Modul im Editor umbenannt. Wird also dieses Modul aktiviert, so wird ein anderes Event aufgerufen. Sind im referenzierten Event, sowie dem Event in welchem das Modul liegt, Parameter mit identischem Namen, so werden diese durchgereicht (Sound Librarian, kein Datum). In diesem Nested Event können nach belieben andere Module eingesetzt werden.



Abbildung 22: Event Sound

5.1.4 Randomisierung

Wie aus den Event und Scatterer Sounds hervorgeht, ist es in FMOD möglich, diverse Module auf eine komplexe Art und Weise ineinander zu verschachteln. Dabei ist es zusätzlich möglich, Einstellungen innerhalb dieser Module unabhängig voneinander zu *randomisieren*.

Unter *Randomisierung* wird in FMOD die zufällige Veränderung eines Wertes innerhalb eines definierten Rahmens verstanden, wie etwa in den bereits erwähnten Optionen der Scatterer Sounds. Doch die recht offensichtlichen Schieberegler sind nicht die einzigen Parameter die verzufällig werden können.

Nahezu alle Werte aller Effekte, zum Beispiel auch die Grenzfrequenz eines Tiefpasses, lassen sich in FMOD randomisieren; durch Drücken der Alt-Taste und der linken Maustaste lassen sich diese zu Drehreglern hinzufügen. Im Deck wird der Reiter *Modulation* hinzugefügt, der die jeweils randomisierten Effekte beinhaltet. Zusätzlich erscheint um den jeweiligen Regler eine dicke Linie, die den randomisierten Bereich darstellt. Im Falle der Multi Sounds werden häufig Volume und Pitch um jeweils ein paar dB und Cent variiert, um die Zahl der Assets größer wirken zu lassen und somit einer Abnutzung entgegenzuwirken. (vgl. Abbildung 23).



Abbildung 23: Randomisierung

5.1.5 Triggerverhalten

Unabhängig der gewählten Modulart, kann im Reiter *Trigger Behavior* zu jedem Modul eingestellt werden, mit welcher Wahrscheinlichkeit es abgespielt wird. Auch dieser Wert lässt sich Randomisieren. Mit dem Würfelsymbol lässt sich diese Wahrscheinlichkeitsänderung aktivieren (Würfel = gelb) und deaktivieren (Würfel = grau).

Das Abspielen eines Moduls kann hier auch an Wertebereiche von möglichen Parametern gebunden werden. Polyphony ist bereits aus den Scatterer Sounds bekannt und erfüllt hier dieselbe Funktion. Unter Time kann eine Zeit eingestellt werden, in der das Modul stummgeschaltet ist, obwohl der Play-Cursor es bereits überstrichen hat.

Tempo hat denselben Effekt, nur dass erst eingesetzt wird, wenn der mitgezählte Notenwert des Events mit dem hier ausgewählten Notenwert zusammenfällt. Dazu muss ein Tempomarker im Event platziert sein. Unter Seek kann ein prozentualer Offset des Playbacks eingestellt (und randomisiert) werden (Sound Librarian, kein Datum).



Abbildung 24: Trigger Behavior

5.1.6 Parameter

Die an dieser Stelle besprochenen *Parameter*, sind diejenigen, die später durch die Game Engine in Echtzeit geändert werden. Dabei sind diese Parameter an jeweils eine Instanz eines Event gebunden; wird dasselbe Event simultan ausgeführt, so sind die Parameter jeder Ausführung eigenständig und unabhängig voneinander.

Neue Parameter können mit einem Klick auf das kleine Plus neben dem Timeline-Reiter hinzugefügt werden. Dabei wird abgefragt, ob der Sound Designer einen komplett neuen Parameter erstellen, oder aber auf einen *Built-In Parameter* zurückgreifen möchte. Letztere werden standardmäßig bei 3D-Objekten errechnet; sie betreffen etwa die Entfernung, Ausrichtung, Höhe, etc. (vgl. Abbildung 25).

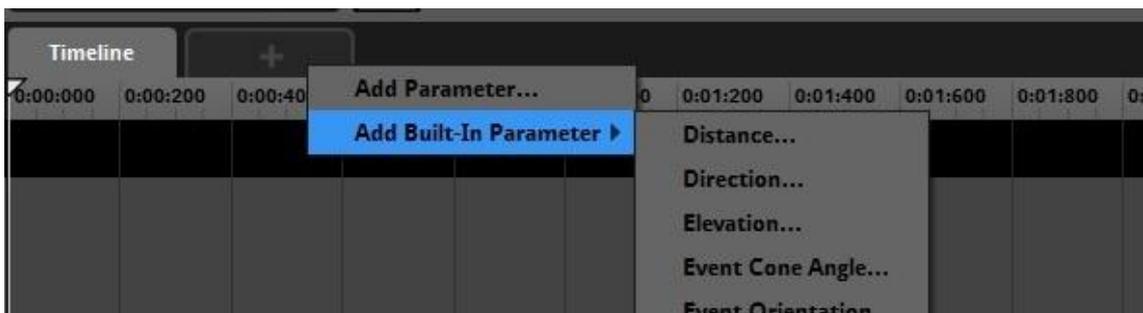


Abbildung 25: Hinzufügen von Parametern

Möchte man zusätzliche Änderungen des Klangbildes erreichen, zum Beispiel die Pegel verschiedener Komponenten eines Events unterschiedlich mit der Entfernung des Listeners zur Schall-

quelle verändern, so können diese beim Erstellen neuer Parameter ausgewählt werden. Der Vorteil hierbei ist, dass kein Mehraufwand für den Programmierer entsteht, da diese Informationen bei 3D Events ohnehin durch FMOD aus der Game Engine bezogen werden.

Für alle Änderungen innerhalb eines Events, bei denen diese nicht geeignet sind, können neue, sozusagen leere Parameter verwendet werden. Diese müssen durch neuen Code des Programmierers verändert werden. Parameter sind in den Reitern neben Timeline aufgeführt. Klickt man auf den Reiter eines Parameters, so verschwindet die Zeitleiste und wird durch eine Leiste ersetzt, die dem gesetzten Wertebereich des Parameters entspricht (dieser Bereich ist nachträglich anpassbar). Es kann eine beliebige Anzahl an Parametern in einem Event verwendet werden (Sound Librarian, kein Datum). Zusätzlich erscheint neben der Transportkonsole ein Drehknopf, der ebenfalls diesen Wertebereich wiederspiegelt. Das Drehen dieses Reglers passt den aktuellen Wert des Parameters an (vgl. Abbildung 26).



Abbildung 26: Mehrere Parameter in FMOD

Ein gewünschter Effekt ist häufig auf unterschiedliche Art und Weise erreichbar, da die Parameter unterschiedlich eingesetzt und kombiniert werden können. Unter anderem können in den Parametern Automationen geschrieben, ihr Triggerverhalten definiert und sogar Assets platziert werden. Beides soll in eigenen Kapiteln besprochen werden. (Automationen durch Parameter sollen dabei in einem Beispiel des Echtzeit-Layering vorgestellt werden.)

5.1.7 Sound-Assets in Parametern

Immer dann, wenn das zeitliche Verhalten eines Assets unwichtig ist und es stattdessen an andere Anforderungen gebunden ist, bietet sich eine Platzierung der Assets innerhalb eines Parameters an. Dafür können Assets, genau wie beim Platzieren auf der Timeline, per Drag-and-Drop platziert, verschoben und ineinander überblendet werden.

Eine klassische Anwendung für dieses Vorgehen ist das Vertonen eines Fahrzeugs, gemäß der Umdrehungen pro Minute des Motors. Hierfür könnte ein Parameter RPM (engl.: rotations per minute) mit entsprechendem Wertebereich definiert werden. Nun können ein Asset für eine niedrige und ein Asset für eine hohe Drehzahl in diesem Parameter platziert werden. Durch einen passenden Crossfade können die Werte zwischen den Extremen abgebildet werden; die Parameter in FMOD sind als Float anwählbar (Kellaway & FMODTV, 2014).

In THE DARK SIDE OF THE MOON wurde das Hacking-Minigame durch Sounds innerhalb eines Parameters umgesetzt. In diesem Minigame gibt es vier Felder; jedem dieser Felder ist ein Sound zugeteilt. Es gilt nun die Sequenz dieser Felder richtig zu Wiederholen. Benötigt wird also ein Parameter `lockpicking`, mit mindestens vier möglichen Werten. Es ist also je eines der Assets an der Stelle des jeweiligen Wertes platziert worden (vgl. Abbildung 27).



Abbildung 27: Sound Assets in einem Parameter, Lockpicking

5.1.8 Echtzeit-Layering

Echtzeit-Layering verfolgt denselben Gedanken wie das bereits vorgestellte statische Layering: verschiedene Schichten werden kombiniert um einen neuen Sound zu kreieren.

Diese Form des Layerings geht noch einen Schritt weiter. Nicht einzelne Sounds werden kombiniert und dann als feste Datei gebounced; stattdessen werden Containermodule, beispielsweise Multi Sounds gelayert. Durch die zufällige Kombination der abgespielten Assets ergibt sich eine Vielzahl an Möglichkeiten. Diese Zahl der unterschiedlichen resultierenden Sounds kann durch weitere Randomisierung zusätzlich erhöht werden. Aus diesem Grund ist Echtzeit-Layering ein gutes Werkzeug um eine größere Varianz zu erzeugen und einer Abnutzung von Sounds entgegenzuwirken. Es sollte bedacht werden, dass durch die separat angelegten Bestandteile der Layer auch der Speicherbedarf höher ist. Daher sollte evaluiert werden, welche Sounds am meisten durch ein Echtzeit-Layering profitieren und in welchen Fällen die erzeugte Varianz ein wenig übertrieben wäre (Vachon, 2009).

Als Beispiel werden an dieser Stelle die Fußschritte aus THE DARK SIDE OF THE MOON betrachtet: Ein einzelner Fußschritt besteht immer aus zwei unterschiedlichen Kleidungsraschelmodulen. Ein Rucksackrascheln tritt mit einer 32-prozentigen Wahrscheinlichkeit auf. Zusätzlich kommt ein vom Untergrund abhängiges Auftrittgeräusch hinzu, sowie die Möglichkeit, dass dieser nass ist. Die Konstellation der Fußschritte auf dem trockenen, herkömmlichen harten Boden der Raumstation ist in Abbildung 28 zu sehen.



Abbildung 28: Fußschritte auf hartem Untergrund

Um auf die unterschiedlichen Untergründe zuzugreifen, werden Parameter benötigt. In THE DARK SIDE OF THE MOON wurde dies durch ein parameterbedingtes Ducking in Form einer Automation umgesetzt. Diese Automation wird im jeweiligen Reiter des Parameters geschrieben²⁸. Im Falle des Standardbodens bedeutet dies: Ist Std Floor = 0, so ist diese Spur mit -80 dB maximal gedämpft und nicht hörbar; ist Std Floor = 1, so erreicht der Kanal den Master mit 0 dB, also ungedämpft (vgl. Abbildung 29). Die anderen Böden wurden ebenfalls so umgesetzt.

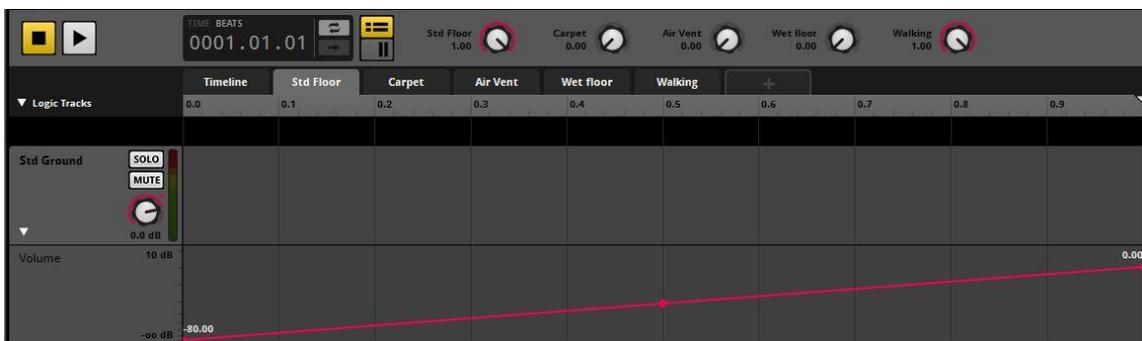


Abbildung 29: Parameterautomation

²⁸ Um eine Automation zu einem Parameter hinzuzufügen, muss über einen Rechtsklick auf seinen Drehregler zunächst eine Volume-Modulation hinzugefügt werden.

Bei diesem Vorgehen können alle Spuren des Events innerhalb der Parameter unabhängig voneinander automatisiert werden. Wird eine Spur gleichzeitig durch zwei Parameter in derselben Regelmöglichkeit automatisiert, beispielsweise beides Mal die Lautstärke, so werden diese Automationen addiert. Es gilt also die Übersicht zu wahren, damit keine ungewünschten Effekte auftreten.

Die Umsetzung der Fußschritte auf diese Weise, bringt den Vorteil mit sich, dass direkt unterschiedliche Konstellationen herbeigeführt werden können, ohne dass weitere Parameter vonnöten sind. Sollen die Fußschritte etwa beim Schleichen nur halb so laut sein? Oder soll etwa normales Gehen nur ein wenig leiser als Rennen sein? Da die Parameter als Float angelegt sind, ist jeder Wert der Automation aufrufbar und auch über die Game Engine ansprechbar. Die Beispiele sind also durch unterschiedliche Werte der Parameter umsetzbar, ebenso wie flüssige Übergänge dazwischen.

Eine weitere Methode, die unterschiedlichen Layer aufzurufen besteht darin, das Triggerverhalten der einzelnen Module anzupassen. Damit der Standardboden bei `Std Floor = 1` mit dieser Methode zu hören ist, muss unter Trigger Behaviour die *Condition* Std Floor eingefügt und der Slider auf 1 gesetzt werden (vgl. Abbildung 30). In diesem Fall finden keine Automationen statt, eine Dämpfung müsste also zusätzlich hinzugefügt werden.

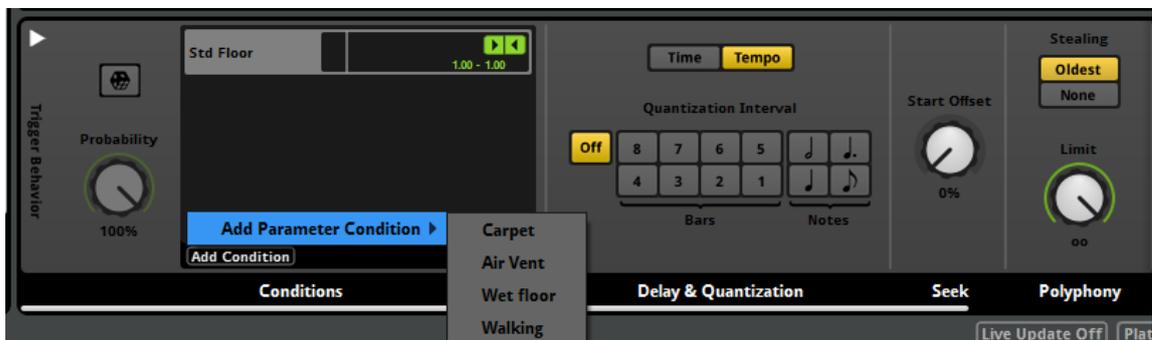


Abbildung 30: Parameter im Triggerverhalten

Man könnte durch die Verwendung des Triggerverhaltens und der Erweiterung des Wertebereichs eines Parameters, die restlichen Parameter obsolet machen. Gegeben seien die drei Untergründe aus THE DARK SIDE OF THE MOON: Standardboden, Teppichboden und Lüftungsschacht. Nun könnte ein Parameter Walking definiert werden, der drei Zustände einnehmen kann. Jedem Untergrund wird nun ein Wert zugeteilt. Soll es für zwei dieser drei Untergründe zusätzlich möglich sein, trocken oder nass zu sein, so ist dies ohne zusätzlichen Parameter möglich. Die beiden Böden die nass sein können sollen, müssen benachbarte Werte einnehmen. Sollen alle drei Untergründe nass sein können, so wird ein weiterer Parameter benötigt. Dies ist darin begründet, dass, wie in Abbildung 30 zu sehen ist, kein zweites Werteintervall für einen bereits gelegten Trigger definiert werden kann.

5.1.9 Logic Track

Einer der großen Unterschiede einer Middleware wie FMOD zu herkömmlichen DAWs und Sequenzern besteht darin, dass durch das Erfüllen von Bedingungen, also einer *Logik*, auf der Zeitachse diverse Aktionen durchgeführt werden können, zum Beispiel Sprünge an andere Stellen.

Hierzu dient der *Logic Track*, welcher sich direkt unter der Timeline befindet. Hier können diverse Marker gesetzt werden, unter anderem:

- Marker
Marker sind grau. Sie können beschriftet werden und dienen dazu, Abschnitte in der Timeline eines Events zu markieren und können durch Transition Marker/Regions angesprungen werden.
- Transition Marker
Transition Marker sind grüne Marker; sie geben ihr Ziel mit dem Wort *to* an. Überstreicht der Playcursor einen Transition Marker und sind alle Bedingungen erfüllt, oder keine Bedingungen definiert, so springt der Playcursor an die Stelle des jeweiligen Markers.
- Transition Region
Transition Regions sind grüne Kästen. Auch sie geben ihr Ziel mit *to* an. Ist die Sprungbedingung erfüllt und befindet sich der Playcursor innerhalb dieser Region, so springt er an den jeweiligen Marker.
- Loop Region
Loop Regions werden als blaues Kästchen dargestellt. Erreicht der Playcursor das Ende der Region und wurde keine Abbruchbedingung gesetzt, so springt er wieder an den Anfang dieser Region.



Abbildung 31: Logic Track und Marker

Durch den Einsatz von diesen Markern ist ein komplexes Audioverhalten möglich. Dabei können, wie bereits angedeutet, Parameter eingesetzt werden. Neben den bereits besprochenen Möglichkeiten eines Parametereinsatzes, können sie also benutzt werden, um in der Timeline zu springen,

zu loopen, oder einen Sound zu beenden. Auch können die vorgestellten Sound Module und sämtliche Randomisierung eingesetzt werden und zusätzlich in Echtzeit gelayert werden. Ein anschauliches, nicht allzu komplexes Beispiel ist der Inventar-Sounds aus THE DARK SIDE OF THE MOON (vgl. Abbildung 32):

Öffnet der Spieler das Inventar, so startet das FMOD-Event. Solange der Spieler sich im Inventar befindet, ist der Parameter `close = 0` und der Bereich innerhalb der Loop Region wird in Dauerschleife abgespielt. Den Multi Sounds in der unteren Spur wurden jeweils unterschiedliche Auslösewahrscheinlichkeiten zugeteilt (ein ähnlicher Effekt wäre auch mit einem Scatterer Sound möglich). Schließt der Spieler das Inventar, so wird `close = 1` gesetzt. Die Bedingung der Transition Region ist somit erfüllt und der Playcursor springt zum Marker Out; das Event klingt aus.

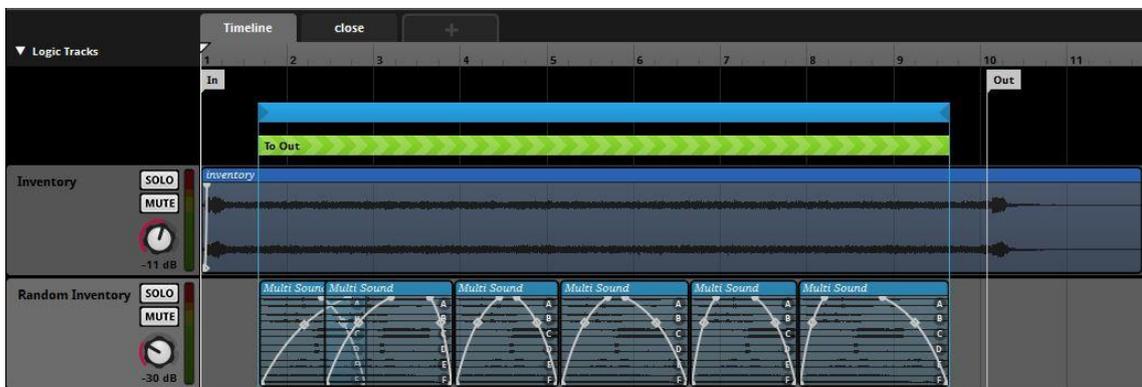


Abbildung 32: Feuersound aus THE DARK SIDE OF THE MOON

5.1.10 Begrenzung der Anzahl getriggelter Events

Manchmal kann es vorkommen, dass ein Event zu schnell hintereinander getriggert wird, oder zu viele Instanzen gleichzeitig aufgerufen sind. Ein Beispiel: Betätigt der Spieler häufig und in kurzer Zeit die Taste für Vorwärtslaufen, so würde es so klingen, als ob der Charakter auf der Stelle steppt. Der Sound, insbesondere des ersten Schrittes, ist an das betätigen dieser Taste gebunden (vgl. Auditory Icon, Response).

Um dies zu verhindern, können in FMOD in den *Event Macros* diverse Werte festgelegt werden (vgl. Abbildung 33):

- Max Instances und Stealing

Max Instances definiert, wie viele Instanzen eines Events gleichzeitig aufgerufen sein können. *Stealing* beschreibt, welches Event beendet wird, sobald eine weitere Instanz, trotz Erreichen der maximalen Anzahl, getriggert wird.

- Cooldown

Cooldown ist diejenige Zeit, die zwischen dem Aufrufen zweier Instanzen mindestens vergangen sein muss (Sound Librarian, kein Datum).

- Priority

Die Priorität legt fest, wie wichtig das Event im Vergleich zu anderen ist. Wird die Zahl der simultan abspielbaren Kanäle überschritten, wird anhand der fünf Prioritätsstufen entschieden, welche Events abgespielt und welche beendet werden (Leenders, 2012). Ein hochpriorisierter Sound wird auch dann nicht zwangsbeendet, wenn er im Vergleich zu Sounds mit einer niedrigeren Priorität sehr leise ist (Firelight Technologies, kein Datum).



Abbildung 33: Event Macros

5.1.11 Snapshots

Um die Kontrolle über das Klangerlebnis des Spieles zu erhöhen, also um Modul- und Buseigenschaften innerhalb des Mixers²⁹ zu verändern, gibt es in FMOD die Möglichkeit, *Snapshots* anzulegen. Zusätzlich können Übergänge zwischen verschiedenen Snapshots definiert werden (Sound Librarian, kein Datum).

²⁹ Hierzu muss in das Mixer-Fenster gewechselt werden (Strg+2).

Um die Snapshots effektiv nutzen zu können, muss vorher eine sinnvolle und für die geplanten Anwendungen praktikable Bus-Struktur angelegt werden. In THE DARK SIDE OF THE MOON sollten bestimmte Charaktersounds beispielsweise verhallt werden (Fußschritte, Schwingen des Arms, etc.) und andere nicht (Statussounds wie etwa das Herzklopfen bei niedriger HP, vgl. Abbildung 34).

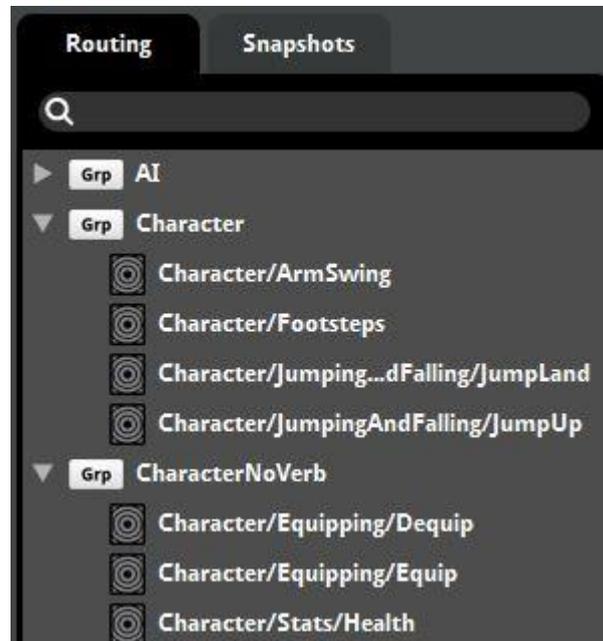


Abbildung 34: Routing der Charaktersounds

In einem Snapshot ist die Änderung zum Default gespeichert, dies bedeutet auch, dass nur die Busse eingespeichert werden, die verändert wurden. Somit ist gewährleistet, dass das System dynamisch verändert werden kann. Nicht vom Snapshot betroffene Einstellmöglichkeiten sind gestrichelt dargestellt. In FMOD gibt es zwei unterschiedliche Arten von Snapshots (Sound Librarian, kein Datum):

a) Overriding Snapshot

Ein Overriding Snapshot überschreibt die Werte, welche von den in ihm gespeicherten abweichen.

b) Blending Snapshot

Blending Snapshots verhalten sich wie Overriding Snapshots, mit dem Unterschied, dass die Einstellungen der Fader summiert werden.

Werden mehrere Snapshots gleichzeitig aufgerufen, so werden auch beide von FMOD durchgeführt. Dabei wird der Snapshot mit der höchsten Hierarchie zuletzt aufgerufen. Die Snapshots sind in ihrer Hierarchie aufgelistet; oben sind die mit der höchsten, unten die mit der niedrigsten.

Klassische Anwendungsbeispiele für Snapshots in FMOD sind etwa ein Ducking, oder das Anlegen verschiedener Hallräume.

In THE DARK SIDE OF THE MOON sollte durch ein Ducking sichergestellt werden, dass der Hauptdialog stets zu hören ist. Hierfür wurden die anderen Geräuschgruppen um einen festen Wert und die statusbezogenen Sprüche vollständig gedämpft. Zusätzlich wurde ein Kompressor auf die Musik- und Atmogruppe gelegt, welcher über ein Sidechain auf den Hauptdialog regiert (vgl. Abbildung 35).



Abbildung 35: Ducking via Snapshot

Auch die unterschiedlichen Hallräume wurden auf ähnliche Weise in THE DARK SIDE OF THE MOON umgesetzt (vgl. Abbildung 36): In den jeweiligen Snapshots wurden entsprechende Hall-Sends gespeichert und jeweils der Hall an die Raumbeschaffenheit angepasst.

Der Hall-Return wurde im Live-Mixing angepasst. Ein flüssiger Übergang zwischen den unterschiedlichen Hallräumen ist essentiell, um eine realistische Abbildung der Spielwelt zu kreieren (Rumsey, 2010). Um dies zu erreichen, wurde jeweils der Master mit einer AHDSR-Modulation versehen. Auf diese Weise wird eine Art Crossfade beim Wechsel zwischen den jeweiligen Snapshots angelegt (die Attack-Zeit beschreibt wie lange es dauert, bis der Snapshot seinen vorgesehenen Wert erreicht hat, die Release-Zeit wie lange er zum Abklingen benötigt).



Abbildung 36: Hallraum via Snapshot

5.1.12 Soundbanks

Damit die Events in eine Game Engine eingebunden werden können, müssen diese zunächst in sogenannte *Banks* eingeteilt³⁰ und auch als bank-Datei *gebildet* werden.

Eine Bank beinhaltet die angelegten Events samt Parameter, das zugrundeliegende Audiomaterial und die dazugehörigen Metadaten. Wird ein Event geladen, so greift es also auf die, in den Banks gelagerten Dateien und Bearbeitungsanweisungen zu. Wird eine Bank geladen, so werden auch alle zugehörigen Daten in den Speicher geladen. Dies hat den Vorteil, dass die Sounds nahezu

³⁰ Rechtsklick auf ein Event → *Assign to Bank*

ohne Latenz wiedergegeben werden können (Leenders, 2012; Firelight Technologies, kein Datum).

Hieraus ergibt sich, dass die Struktur der Banks direkt einen Einfluss auf die Performance des Spieles haben kann. Es ist bei größeren Produktionen also sinnvoll, die Events derart in Banks einzuteilen, dass nicht benötigte Sounds nicht permanent geladen sind, sondern beispielsweise nach Level aufgeteilt sind und zwischen den Level, während des Ladebildschirms, nach Bedarf ausgetauscht werden.

Unter den *Preferences* können im Reiter *Build* genaue Einstellungen bezüglich des Exports vorgenommen werden. Dazu zählen unter anderem die Ziel-Directory und Plattform, sowie das Output Format (vgl. Abbildung 37).

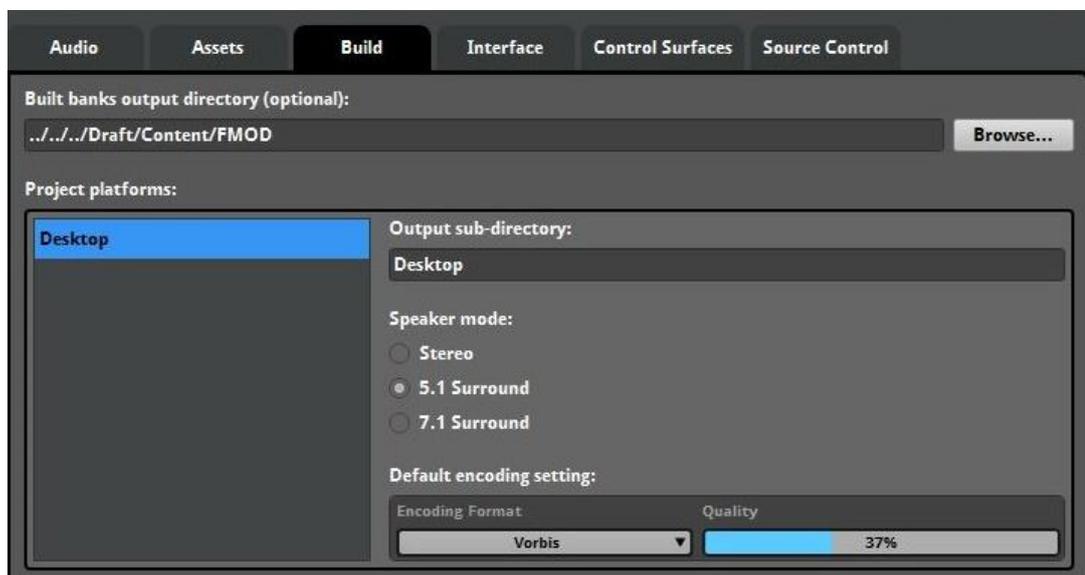


Abbildung 37: Build Preferences

Nach jeder Änderung in FMOD müssen diese Banks neu exportiert werden, da sie immer als eine Art Momentaufnahme zur Zeit des Builds zu verstehen sind.

5.1.13 Live Update

Sind die Events, Snapshots und Parameter in die Game Engine eingebunden, so kann mittels einer Middleware eine klangliche Anpassung, also ein Art Mix, während des Spielens in Echtzeit vorgenommen werden.

In FMOD ist diese Funktion unter dem Namen *Live Update* rechts unten zu finden. Ist diese Option aktiviert, so verbindet sich FMOD beim Start des Spiels in der Game Engine mit dieser und erlaubt es, sämtliche Pegel und Effekte anzupassen. Das Hinzufügen und Entfernen von Assets und Events ist in diesem Live Modus jedoch nicht möglich (Sound Librarian, kein Datum).

Wurde während einer Live-Mix-Session etwas verändert, so ist es ebenfalls notwendig die Banks neu zu builden. Ein Live Update hat keine Auswirkung auf bereits exportierte Banks. Wird nicht neu gebildet, so sind die Änderungen nicht im Spiel enthalten.

5.2 Abschluss der Vertonung

Sind alle Events innerhalb der Middleware umgesetzt, so ist es ratsam, eine Liste mit sämtlichen Events, deren Parametern und einer Beschreibung beider anzufertigen. Selbst wenn diese Beschreibungen dem Sound Designer trivial erscheinen mögen, muss bedacht werden, dass dieser zwar viel Zeit und Gedanken in den Aufbau und die Struktur der Session investiert hat, jedoch nicht davon ausgegangen werden darf, dass der Audio-Programmierer, oder ein anderes Mitglied des Developer-Teams, intuitiv die Bedeutung der Events und Parameter, geschweige denn ihren richtigen Einsatz erfasst. Durch eine solche Liste wird sichergestellt, dass die Sounds im Sinne des Sound Designers eingebunden werden.

Das Wiedergabesystem muss neben der Einstellung in den Preferences der Middleware, auch in der Game-Engine definiert werden. In der Unreal Engine 4 ist dies beispielsweise unter Projekt Einstellungen → Plug-Ins möglich.

Zusätzlich sollte im Hinterkopf behalten werden, dass unter PC-Spielern Kopfhörer und Headsets sehr beliebt sind. Es sollte also darauf geachtet werden, dass die Vertonung – insofern das Spiel auf einer Plattform erscheinen wird, welche Kopfhörer zulässt – auch auf diesen ausgewogen klingt, selbst wenn dies Kompromisse bedeuten kann.³¹ Als alternative Möglichkeit kann angegeben werden, unter welchen Bedingungen das beste Hörerlebnis erzielt wird, oder je einen eigenen Mix für Lautsprecher und Kopfhörer angelegt werden. Es sollte auch bedacht werden, dass manches Userequipment eigenständig einen Kopfhörermix erzeugt, beispielsweise Surround-Kopfhörer. Liefert die Game Engine nun ebenfalls einen Kopfhörermix, so ist offensichtlich, dass das Resultat weder im Sinne des Entwicklers, noch optimal ist. Da eine „Inter-System Awareness“ noch nicht wirklich existiert, sollte entweder darauf hingewiesen werden, dass ein Kopfhörermix direkt vom Spiel kommt, oder aber ein *spatial* Kopfhörermodus in den Optionen auswählbar sein (Digenis, 2015, S. 3). Es muss davon ausgegangen werden, dass der Spieler den Ursprung des Problems nicht kennt und im schlimmsten Fall sogar davon ausgeht, dass sein suboptimales Hörerlebnis auf einen Fehler seitens der Entwickler zurückgeht.

³¹ Das Hörerlebnis bei Kopfhörern unterscheidet sich von dem mit Lautsprechern. Dies ist zurückzuführen auf: Fehlenden Raumklang des Abhörraumes; fehlende interaurale Pegel- und Laufzeitunterschiede; einen veränderten Abhörwinkel (Stereo: 60° Lautsprecher, 180° Kopfhörer) (Digenis, 2015).

Ob ein Sound funktioniert, kann erst beim Spielen des Spiels evaluiert werden. Auch wenn es mitunter schmerzlich sein kann, einen auf dem Papier und an sich handwerklich sehr guten Sound zu verwerfen. Nichtsdestotrotz muss ein Sound Designer auf eine solche Situation gefasst sein. Dies kann in der Regel erst in einem sehr späten Stadium einer Produktion überprüft werden. Deshalb ist es umso wichtiger, Playtests so früh wie möglich abzuhalten. Dabei bietet sich insbesondere ein heuristisches Verfahren an. Hierbei sind lediglich wenige Testpersonen notwendig. Ziel eines solchen Tests ist es nicht, die endgültige Wirkung festzustellen; vielmehr sollen Trends ausgemacht werden und mögliche Problemstellen frühzeitig erkannt werden. Zusätzlich ist es eine wichtige Fähigkeit des Sound Designers, sich selbst beim Spielen in zwei Rollen versetzen zu können: Einmal die Rolle des Neulings, der noch nie Kontakt mit dem Spiel hatte und zum anderen die Rolle des Spielers, der das Spiel bereits exzessiv gespielt hat (O'Donnell, 2002).

6 Fazit

Durch die, in dieser Arbeit diskutierten Aspekte der Computerspielvertonung ist es nun also möglich, eine Vertonung zu kreieren, die nicht nur gut klingt, sondern auch funktional ist und die User Experience insgesamt unterstützt, sowie die Usability verbessert. Es ist ersichtlich, dass eine Computerspielvertonung, bedingt durch unter anderem ihren dynamischen Aufbau, einen überaus komplexen Rahmen einnehmen kann.

Das eigentliche Handwerk der Schaffung der Sounds und der Implementierung dieser unterscheiden sich häufig nur in Details; welche Middleware wurde benutzt, was sind die Geheimtricks eines Foley-Artist, um nur ein Paar zu nennen. Dies bedeutet, dass die Überlegungen aus dieser Arbeit auf andere Genres übertragen werden können. Denn die Entscheidungen aus der Vorproduktion sind es, welche die Natur der Sounds und den Klang des letztlichen Spieles entscheiden und bestimmen, welche Klänge überhaupt produziert werden müssen.

Durch die im Leitfaden analysierten Arbeitsschritte und deren Bestandteile – praktischer, wie theoretische Natur – ist eine zielgerichtete, verbesserte Kommunikation und effizientere Arbeitsweise möglich. Dabei sind die Konzepte nicht als Einschränkung der Kreativität zu verstehen, sondern als Hilfsmittel die Vision des Sound Designers in eine angemessene und funktionale Gestalt zu übersetzen.

Ein Blick auf Nischengenres kann sich lohnen, wie durch die Betrachtung des Survival-Genres gezeigt wurde. Bedingt durch die genreüblichen Mechaniken entstehen besondere Anforderungen an die Vertonung. Durch diese kann es notwendig sein, allgemein akzeptierte Konventionen zu überschreiten und in neue Gefilde der akustischen Umsetzung aufzubrechen: Während es weit verbreitet ist, beispielsweise nur einen einzigen Sound für das Aufsammeln von Items zu verwenden, kann dies in Survival-Games zu einer ungewöhnlich hohen Abnutzung und letztlich einem Immersionseinbruch führen. Es ist hier also erforderlich, zum Beispiel mit einer Varianz zu experimentieren. Genauso ist es sehr wichtig andere Spiele zu analysieren; sowohl innerhalb, als auch außerhalb des favorisierten Genres.

Die Erkenntnisse aus derlei Versuchen und Analysen, können – nach einer Anpassung – auf Spiele anderer Genres übertragen werden. Somit ist eine stetige Synergie und Weiterentwicklung des Audiobereichs für Spiele möglich.

Verweise

- Ament, V. T. (2009). *The Foley Grail : the art of performing sound for film, games and animation*. Burlington, MA: Focal Press.
- Beam Team Pty Ltd. (n.d.). *About Us | Beam Team Games*. Retrieved August 8, 2017, from <https://beamteamgames.wordpress.com/about/>
- Bender, T., & Wulff, H. J. (2012, April 1). *ADR - Lexikon der Filmbegriffe*. Retrieved August 7, 2017, from <http://filmlexikon.uni-kiel.de/index.php?action=lexikon&tag=det&id=61>
- Berndt, A. (2010). Diegetic Music: New Interactive Experiences. In M. Grimshaw, *Game Sound Technology and Player Interaction: Concepts and Developments* (pp. 60-71). Hershey: Information Science Reference (IGI Global).
- Brenna, H. (2015, Februar 23). *Dying Light and the survival renaissance*. Retrieved Juli 30, 2017, from <https://www.vg247.com/2015/02/23/dying-light-and-the-survival-renaissance/>
- Burford, G. (2015, Februar 4). *Most Survival Games Have Problems That S.T.A.L.K.E.R. Solved Long Ago*. Retrieved Juli 27, 2017, from <https://www.kotaku.com.au/2015/02/most-survival-games-have-problems-that-stalker-solved-long-ago/>
- Cancellaro, J. (2006). *Exploring Sound Design for Interactive Media*. New York: Thomson Delmar Learning.
- CBS Sunday Morning. (2015, Februar 22). *A foley artist sounds off [YouTube]*. Retrieved August 1, 2017, from <https://www.youtube.com/watch?v=C8NCxT7XLW8>
- CD PROJEKT RED [GameSpot]. (2016, August 30). *Creating The Sound - The Witcher 3: Wild Hunt Official Developer Diary [YouTube]*. Retrieved from <https://www.youtube.com/watch?v=J-jogZgwf3M>
- Chadwick, E. (2015, August 18). *Art Bible*. Retrieved Juli 29, 2017, from http://wiki.polycount.com/wiki/Art_Bible
- Cullen, M. (2016, April 7). *Basics of Sound Design for Video Games*. Retrieved August 10, 2017, from <http://frost.ics.uci.edu/ics62/BasicsofSoundDesignforVideoGames-MichaelCullen.pdf>
- Day Z. (2017, August 7). Retrieved from <https://de.wikipedia.org/wiki/DayZ>
- Dickreiter, M. (2014). *Handbuch der Tonstudiotchnik* (8 ed., Vol. I). Berlin: De Gruyter.
- Dickreiter, M. (2014). *Handbuch der Tonstudiotchnik* (8 ed., Vol. II). Berlin: De Gruyter.
- Digenis, A. (2015). Challenges of the Headphone Mix in Games. *AES: 56th International Conference: Audio for Games* (pp. 1-5). London: Audio Engineering Society. Retrieved Juli 12, 2017, from <http://www.aes.org/e-lib/browse.cfm?elib=17589>
- Digenis, A. (2015). Challenges of the Headphone Mix in Games. *AES: 56th International Conference: Audio for Games* (pp. 1-5). London: Audio Engineering Society.
- Endnight Games. (n.d.). *FAQ - The Forest*. Retrieved August 8, 2017, from <http://survivetheforest.com/faq/>
- Firelight Technologies. (n.d.). *Virtual Voice System*. Retrieved August 16, 2017, from <http://www.fmod.org/docs/content/generated/overview/virtualvoices.html>
- Firelight Technologies. (n.d.). *Working with Banks*. Retrieved August 15, 2017, from https://www.fmod.org/docs/content/generated/engine_ue4/banks.html
- Fleischmann, M. (2017, Juli). HUD aus The Dark Side of the Moon (Screenshot).

- Fleischmann, M., & Wagner, M. (2017, Juni 7). *GDD_DSoM v7*. Stuttgart.
- Flückiger, B. (2002). *Sound design : die virtuelle Klangwelt des Films* (2. ed.). Marburg: Schüren.
- GamerSpawn. (2010, Juli 29). *Red Dead Redemption - Soundtrack Behind the Scenes [YouTube]*. Retrieved August 10, 2017, from <https://www.youtube.com/watch?v=vEsknPy5rvg>
- Gaver, W. W. (1997). Auditory Interface. In M. Helander, T. Landauer, & P. Prabhu (Eds.), *Handbook of Human-Computer Interaction* (2 ed., pp. 1003-1042). Amsterdam: Elsevier Science Ltd.
- Hinrichsen, S.-T. (2015, Mai 28). *Wissen: Parabolspiegel-Mikrofone*. Retrieved August 10, 2017, from <http://fieldrecording.de/2015/05/28/parabolspiegel-mikrofone/>
- Hinterland Studio Inc. (n.d.). *Our Team*. Retrieved August 8, 2017, from <http://hinterlandgames.com/studio/our-team/>
- Horowitz, S., & Looney, S. (2014). *The essential guide to game audio: the theory and practice of sound for games*. New York: Focal Press.
- Ingwersen, S. (2005). Sonifikation - Zwischen Information und Rausch. In H. Segeberg, & F. Schätzlein (Eds.), *Sound : zur Technologie und Ästhetik des Akustischen in den Medien* (pp. 332-346). Marburg: Schüren.
- iZotope Inc. (2014, September). *Using a Spectrogram to Identify Audio Problems*. Retrieved from <https://www.izotope.com/en/community/blog/tips-tutorials/2014/09/using-a-spectrogram-to-identify-audio-problems.html>
- Jørgensen, K. (2006). On the Functional Aspects of Computer Game Audio. *Proceedings of the Audio Mostly Conference 2006* (pp. 1-5). Piteå: Interactive Institute, Piteå, Sweden. Retrieved August 2, 2017, from <http://bora.uib.no/bitstream/handle/1956/6734/paper-KJorgensen.pdf?sequence=1&isAllowed=y>
- Kellaway, S., & FMODTV. (2014, Oktober 13). *FMOD Studio for UE4 Video 6 - Dynamic Car Engine [YouTube]*. Retrieved Juli 30, 2017, from <https://www.youtube.com/watch?v=BI7EwpEIHuY>
- KPBS News. (2008, November 25). *Foley Artist Explains Sound Effects [YouTube]*. Retrieved August 2, 2017, from <https://www.youtube.com/watch?v=szyht9jc8PE>
- Lane, R. (2013, Juli 5). *Virtual Selection: The Rise of the Survival Game*. Retrieved August 3, 2017, from <http://www.ign.com/articles/2013/07/05/virtual-selection-the-rise-of-the-survival-game>
- Leenders, M. J. (2012). *Sound für Videospiele : besondere Kriterien und Techniken bei der Ton- und Musikproduktion für Computer- und Videospiele* (1 ed.). Marburg: Schüren.
- Lennartz, S. (2009, Juni 24). *Moodboards – das Design vor dem Design*. Retrieved Juli 29, 2017, from <https://www.drweb.de/magazin/moodboards-design-entwur/>
- Marks, A. (2017). *Aaron Marks' Complete Guide to Game Audio: For composers, sound designers, musicians and game developers* (3 ed.). Boca Raton, FL: CRC Press.
- Obst, H. (n.d.). *Machinegun-Effekt*. Retrieved August 6, 2017, from <https://www.releasetime.de/machinegun-effekt/>
- O'Donnell, M. (2002, Mai 20). *Producing Audio for Halo*. Retrieved August 1, 2017, from http://www.gamasutra.com/resource_guide/20020520/odonnell_pfv.htm
- Phillips, W. (2014). *A Composer's Guide to Game Music*. Cambridge, MA: MIT Press.

- Richartz, J. (2016). *Spiele entwickeln mit Unreal Engine 4 : Programmierung mit Blueprints : Grundlagen & fortgeschrittene Techniken*. München: Hanser.
- Rumsey, F. (2009, Juni). Sound and Music for Games: Prokofiev to Pac-Man to Guitar Hero. *Journal of the AES*, 6(57), pp. 455-459. Retrieved Juli 12, 2017, from <http://www.aes.org/e-lib/browse.cfm?elib=14828>
- Rumsey, F. (2010, April 4). The Art and Business of Game Audio. *Journal of the AES*, 58(5), pp. 332-335. Retrieved Juli 12, 2017, from <http://www.aes.org/e-lib/browse.cfm?elib=15257>
- Rumsey, F. (2010, April 4). The Art and Business of Game Audio. *Journal of the AES*, 58(5), pp. 332-335.
- Schmidt, B. (2003). Interactive Mixing of Game Audio. *Convention Paper 5857* (pp. 1-6). New York: Audio Engineering Society.
- Schütze, S., & FMODTV. (2012, Oktober 12). *FMOD Studio Tutorial 02 - Project Setup*. Retrieved August 10, 2017, from <https://www.youtube.com/watch?v=BEQFN4bihjA>
- Smith, G. (2014, Oktober 20). *Survival Games Are Important*. Retrieved Juli 25, 2017, from <https://www.rockpapershotgun.com/2014/10/20/survival-games-are-important/>
- Sound Librarian. (kein Datum). *FMOD Studio User Manual (FMOD Studio V1.08.08)*. Retrieved Juni 5, 2017
- Sykora, D. (2016, November 8). *Audio for Unity 5: Survival Shooter (5/5) - Profiling and Final Touches [YouTube]*. Retrieved August 12, 2017, from <https://www.youtube.com/watch?v=T3SmLEiSUPM&index=5&list=PLp4vT3ssm5SUGZiG7dPaIfan1c0khdTKS>
- Thornton, M. (2007, Februar). *Using Fades & Crossfades*. Retrieved August 5, 2017, from <https://www.soundonsound.com/techniques/using-fades-crossfades>
- Thudium, S. (2015, April 29). *The Witcher 3 Screenshots*. Retrieved August 1, 2017, from <http://www.areagames.de/games/the-witcher-3-wild-hunt/news/vier-neue-screenshots-witcher-3-wild-hunt-135852/>
- Vachon, J.-F. (2009). Avoiding Tedium - Fighting Repetition in Game Audio. *AES 35th International Conference: Audio for Games* (pp. 1-12). London: Audio Engineering Society. Retrieved Juli 12, 2017, from <http://www.aes.org/e-lib/browse.cfm?elib=15158>
- Vachon, J.-F. (2009). Avoiding Tedium - Fighting Repetition in Game Audio. *AES 35th International Conference: Audio for Games* (pp. 1-12). London: Audio Engineering Society.
- Wagner, M. (2017). *Audio_Asset_List_v2*. Stuttgart.
- Wagner, M. (2017). *DarkSide_Script*. Stuttgart.
- Weiss, P. P. (2015). *Wenn Design die Materie verlässt : Sound. Das Design der Emotionen, der Imagination und der Lebendigkeit*. Norderstedt: BoD - Books on Demand.