WELK
ENTWICKLUNG EINES
POINT & CLICK ADVENTURES

WELK
ENTWICKLUNG EINES POINT & CLICK ADVENTURES

Intermediate Project Hauptthema
März 2012
Volker Zerbe

Audiovisuelle Medien
Hauptprüfer: Prof. Björn Bartholdy
Nebenprüfer: Prof. Andreas Wrede

Köln International School of Design

INHALT

6/7 **Einleitung**

8 **Zielsetzung**

9 **Das Genre**

10/12 **Storytelling in Non-Linearen Medien**

13 **Die Geschichte**

14/15 **Planung der Interaktion**

16/17 **Moodboards**

18/19 **Settings**

- 20–23 Layout der Wohnung
- 24-33 Zeichnen der Räume & Kolorierung
- 32/34 Animation der Räume
- 35 Erstellung der Grafiken

36/37 **Charakterentwicklung**

- 38/39 Charakter Moodboards
- 40–43 Charakterskizzen
- 44/45 Finalisierung des Charakters
- 46/47 Grundlagen Charakteranimation
- 48/49 Technische Umsetzung der Animation
- 50–55 Arbeitsschritte Animation

56–59 **Items & Interfaces**

60/61 **Soundeffekte & Musik**

62/63 **Arbeite mit Game-Engines/Visionaire Studio**

- 64/65 Anlegen von Charakteren
- 66/67 Mauszeiger erstellen
- 68/69 Schriften
- 70–75 Kulissen
- 76–79 Anlegen des Inventars
- 80/81 Items

82–87 **Screenshots des Spiels**

88 **Fazit/Ausblick**

89 **Quellen**

90 **Assertion**

EINLEITUNG

Innerhalb der Computerspielebranche vollzieht sich zur Zeit ein Paradigmenwechsel. Mitte der 90er Jahre und Anfang des neuen Jahrtausends wurden erfolgreiche Spiele in der Regel nur von den größeren marktbestimmenden Firmen entwickelt. Nun schaffen immer mehr Indie-Entwicklerstudios, unabhängige Teams bestehend aus nur wenigen Leuten, mit ihren Spielen weltweite Erfolge. Spiele wie *„Machinarium, Amanita Design, 2009“*, *„Amnesia, the Dark Descent, Frictional Games, 2010“*, *„Limbo, Playdead, 2010“*, *„Edna bricht aus, Deadalic Entertainment, 2008“* und *„Trauma, Krystian Majewski, 2011“* stehen nur exemplarisch für die Bandbreite an international erfolgreichen Indie-Spielen.

Im Gegensatz zur stark industrialisierten Branche der *„Mainstream-Games“,* in der bestehende Spielmechaniken nur noch wenig weiterentwickelt werden und sich die Verbesserungen oft auf grafische Weiterentwicklungen beschränken, stechen die Beispiele der Indie-Entwickler durch besonders individuelle visuelle Konzepte und innovative Spielmechaniken hervor.

Doch die Indie-Entwicklerbranche besteht nicht nur seit diesem Jahrtausend. Anfang der 90er Jahre wurden viele der heutzutage als Mainstream gehandelten Entwicklerstudios von nur wenigen Entwicklern gegründet. Die damals entwickelten Spiele legten dabei mit ihrer Spielmechanik den Grundstein für viele heutzutage populäre Spielegenres. So wurde der als Klassiker gehandelte Spieletitel *„Another World, Eric Chahni, 1991“* von nur einer Person entwickelt. Das Action-Adventure um einen Professor, der sich nach einem fehlgeschlagenen Experiment auf einem unbekannten Planeten behaupten musste, setzte zu dieser Zeit in Sachen cineastischer Präsentation, Storytelling und rotoskopierter Animationen Maßstäbe. Der wirtschaftliche Erfolg hielt sich jedoch in Grenzen.

Was macht also den Erfolg von Indie-Games heutzutage aus? Computerspiele werden nun deutlich stärker in unserer Kultur wahrgenommen. Künstlerisch interessante Konzepte involvieren nun auch Menschen, die vorher weniger Zugang zu Spielen hatten. Ein weiterer wichtiger Faktor sind die erleichterten Vertriebswege durch das Internet. Entwickler müssen nicht mehr nach Publishern für die Serienproduktion ihrer Spiele suchen, sie können ihre Produkte selbst über Online-Vertriebsplattformen anbieten.

Die Vielzahl an immer leichter zu bedienender Grafik- und Autorensoftware ermöglicht dabei einer größeren Bandbreite an Entwicklern ihre eigenen Spielvisionen umzusetzen. Mittlerweile braucht man durch speziell entwickelte Game-Engines immer weniger spezielle Programmierkenntnisse, um eigene Spiele zu entwickeln. Im Bereich der Point & Click Adventures hat sich dabei besonders die von der siteface UG entwickelte Point & Click Engine *„Visionaire Studio“* hervorgetan. Das deutsche Indie-Entwicklerteam Daedalic Entertainment schaffte mit dem auf dieser Engine entwickelten Adventure *„Edna bricht aus“* den Durchbruch und erzielt mit Neuentwicklungen wie *„Deponia“* hohe Verkaufszahlen im ansonsten gegenüber dem Konsolenmarkt eher zurückgegangenen Verkaufszahlen von PC-Spielen.

Machinarium
Amanita Design
Tschechien 2009

Amnesia The Dark Descent
Frictional Games
Schweden 2010

Limbo
Playdead
Schweden 2010

Edna bricht aus
Deadalic Entertainment
Deutschland 2008

Trauma
Krystian Majewski
Deutschland 2011

Another World
Eric Chahi
Frankreich 1991

ZIELSETZUNG

Inspiriert durch den Erfolg und die Qualität der zuvor genannten Indie-Spiele entschied ich mich, ebenfalls auf Basis der Visionaire Engine den Prototypen für ein eigenes Point & Click Adventure zu entwickeln. Nachdem ich mit dem Spiel „Drunk“ erste Erfahrungen mit der Entwicklung von Point & Click Adventures auf Flash-Basis gesammelt hatte, wollte ich in einem zweiten Ansatz ein weiteres Spiel entwickeln. Als besonderes Merkmal sollte dabei ein malerischer, künstlerischer Stil kombiniert mit einer starken Story entwickelt werden. Eine düstere, depressive Atmosphäre in Stil der Filme von David Lynch mit realistischen Settings würde im Kontrast zu den meist comichaften Adventures stehen. Vielen Perspektivwechseln sollten visuell ein filmisches Erlebnis bieten.

Um das Projekt in der angemessenen Zeit umsetzen zu können, wurde der Umfang des Spiels auf den Prolog der Story beschränkt. Das Gameplay dieses Prototypen sollte jedoch schon vollständig funktionieren und auf audiovisueller Basis möglichst dem fertigen Spiel entsprechen.

Die Intention dieser Dokumentation ist es, anhand eines Beispiel-Projekts ein Leitfaden für die Entwicklung von eigenen Adventures zu sein. Die einzelnen Arbeitsschritte in den verwendeten Programmen werden detailliert erläutert und mit theoretischem Hintergrundwissen aus dem Gamedesign ergänzt. Damit der Umfang jedoch übersichtlich bleibt, wird ein gewisses Grundverständnis der verwendeten Software vorausgesetzt. Verweise auf entsprechende Online-Tutorials dienen dabei als Ergänzung zu den beschriebenen Arbeitstechniken.

DAS GENRE

Das Genre der Adventure-Spiele zeichnet sich dadurch aus, dass die Spieler eine unbekannte Umgebung erkunden und mit ihr interagieren müssen, damit sich die Geschichte des Spieles entfaltet. Die Spielfigur wird dabei in der Regel aus der 3. Person, also mit einer hinter oder neben der Spielfigur positionierten Kamera, betrachtet und gesteuert. Von allen Spielegattungen ähnelt das Genre der Adventure-Spiele somit in ihrer Geschichtenerzählung und Inszenierung am ehesten dem Film.

Point & Click Adventures stellen im Adventure-Genre eine Spezialität dar, indem Sie ihr Gameplay alleine auf die Steuerung mit einem Mauszeiger reduzieren. Das haptische Feedback des Gameplays ist also, im Gegensatz zu Spielen die ein komplexes Buttonlayout zur Steuerung verwenden, simpel. Die Spiele konzentrieren sich aus dem Reiz, den der Spieler aus den Interaktionen mit der Umgebung und anderen Spielfiguren zieht. Dabei spielt die Entwicklung einer spannenden Geschichte mit interessanten Charakteren eine besondere Rolle.

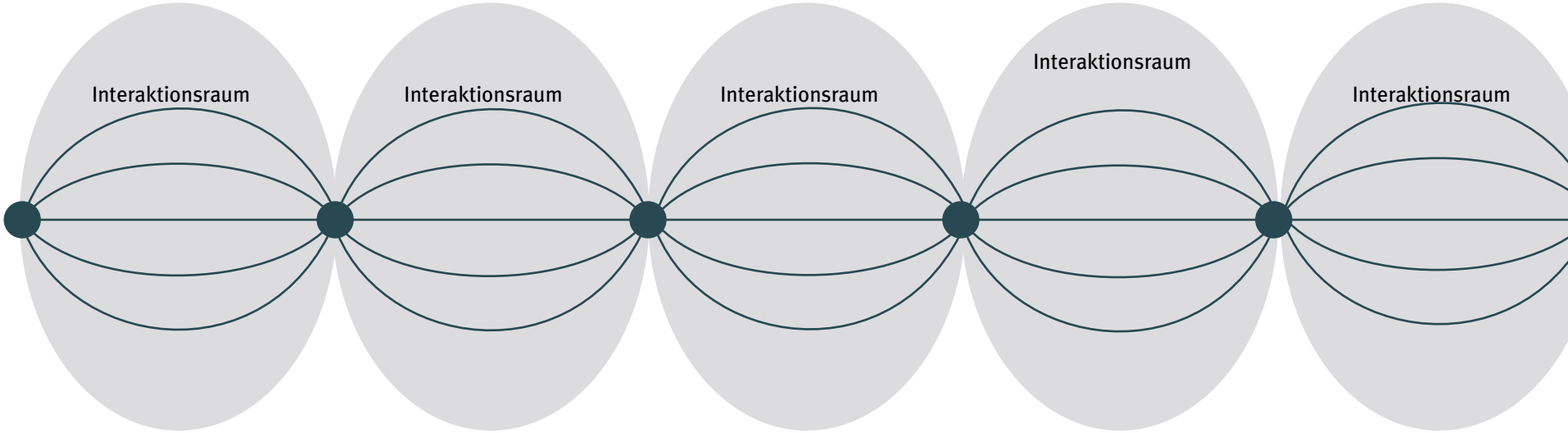
STORYTELLING IN NON-LINEAREN MEDIEN

In Interaktiven Medien habe die Spieler Interaktionsmöglichkeiten, wodurch besondere Anforderungen an das Stroytelling gestellt werden. Jede Geschichte benötigt bestimmte Handlungen, die als Auslöser für den Fortschritt der Geschichte fungieren.
In einem linearen Medium, wie dem Film, hat der Autor die volle Kontrolle darüber welche Information die Betrachter zu welcher Zeit bekommen. In einem non-linearen Medium können die Spieler jedoch weitestgehend selbst bestimmen, wo sie sich zu welcher Zeit befinden. Um jedoch trotzdem eine Geschichte mit einem dramaturgischen Ablauf zu entwickeln, müssen bestimmte Ankerpunkte der Story bestimmt werden. Diese stehen für *Schlüsselmomente* in einer Geschichte, die als Auslöser für weitere Handlungen dienen.

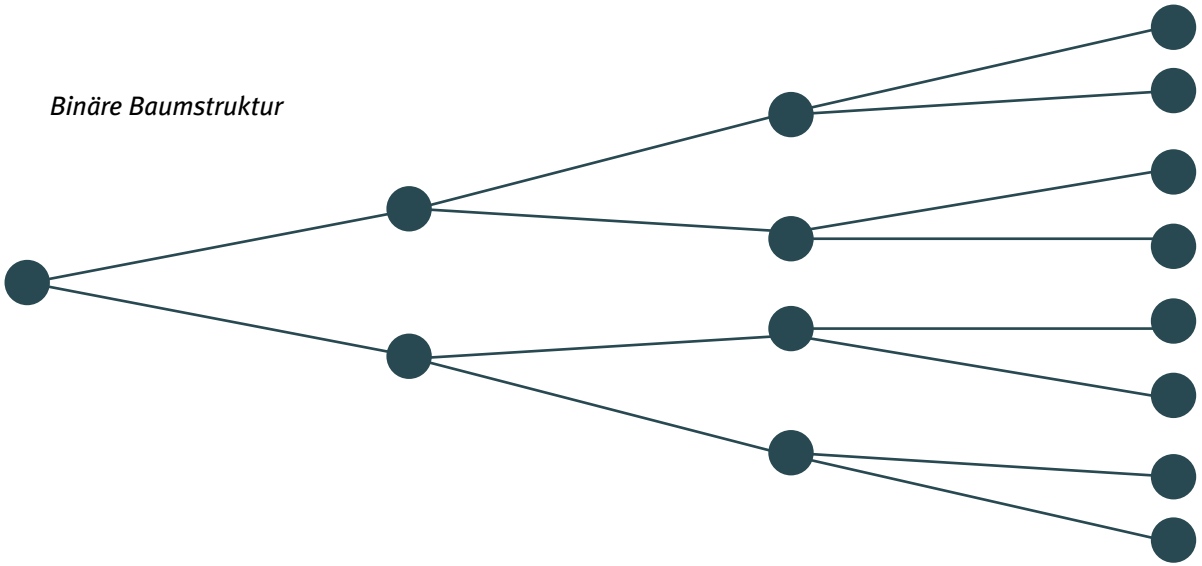
Die Spieler arbeiten also während des Spiels auf die Schlüsselmomente der Geschichte hin. Der Aktionsraum zwischen diesen Schlüsselmomenten wird durch die Interaktion des Spielers bestimmt, man spricht dabei auch von *Interaktionsräumen*. Neben den Ankerpunkten gilt es also auch, diese Interaktionsräume für die Spiele zu bestimmen. Dabei ist es wichtig, dass die Interaktionsräume den Spielern genügend Aktionsfreiheit bieten, um nicht in eine rein lineare Erzählung zu verfallen. Im einfachsten Modell spricht man dabei von sogenannten *„Collapsing Branches“*. Die Story startet an einem bestimmten Punkt und gibt den Spielern ein Ziel vor. Von da ab ist es den Spielern innerhalb des Interaktionsraumes überlassen, wie sie das nächste Storyziel erreichen. Die Geschichte spaltet sich also in verschiedene Nebenstränge auf. Ist das Storyzeil erreicht, laufen die Nebenstränge wieder in einem Schlüsselmomente der Story zusammen, anschließend geht die Story wieder in den nächsten Interaktionsraum über.

Dies ist das einfachste Modell non-linearer Erzählung. Komplexere Modelle berücksichtigen die Handlungen der Spieler innerhalb der Interaktionsräume und führen je nach dem zu unterschiedlichen, parallel liegenden Schlüsselmomenten. Die Spieler können also durch ihre Handlungen den Ablauf der Story beeinflussen. Wird diese Struktur konsequent fortgeführt, wachsen die Handlungsstränge innerhalb der Geschichte exponentiell, man spricht von einer *binären Baumstruktur*.

Collapsing Branches



Binäre Baumstruktur



Eine derartige Geschichte zu entwickeln ist zu komplex. In der Regel ist die Geschichte in Spielen meist eine Kombination des Modells der *Collapsing Branches* mit Elementen des *Binären Baumes*. Ein gutes Beispiel dafür ist das Open-World-Spiel *„Grand Theft Auto 4, Rockstar North, 2006“*. Hier schlüpfen die Spieler in die Rolle eines Gangsters und können Aufträge von verschiedenen Gangs annehmen. Die Reihenfolge, in denen die Spieler die Aufträge erledigen, ist ihnen selbst überlassen - es handelt sich dabei um die zuvor beschriebenen Interaktionsräume. Damit sich jedoch neue Aufträge und damit der Fortschritt der Story erschließt, müssen bestimmte Aufträge abgeschlossen sein. Die Interaktionsräume laufen also wieder in einem *Schlüsselmoment* zusammen.

An einer Stelle im Spiel wird die Hauptfigur vor die Wahl gestellt, eine storyrelevante Figur entweder zu töten oder am Leben zu lassen. Je nach Entscheidung des Spielers ändert sich somit der weitere Verlauf der Story, sie teilt sich somit kurzzeitig in einen *Binären Baum* auf.

Für die Entwicklung der vorliegenden Story mussten also zunächst die Schlüsselmomente der Geschichte definiert werden. Es wurde der grobe Ablauf der Story entwickelt. Inhaltliche Inspirationen lieferten dabei die Geschichten von Filmen wie *„The Machinist, Brad Anderson, Spanien 2004“* oder *„Eraserhead, David Lynch, 1977“*. Anschließend wurden die einzelnen Kapitel der Story ausformuliert. Da für den Prototypen nur der Prolog der Story umgesetzt werden sollte, wurde nur das erste Kapitel vollständig geschrieben.

Die Story beginnt an einem festen Punkt – die Spielfigur startet an ihrem Bett. Anschließend kann der Spieler die Wohnung frei erkunden, er betritt somit einen Interaktionsraum. Mit dem Fund des Zettels in der Küche erhalten die Spieler eine Aufgabe, die für einen weitere Storyfortschritt zu lösen ist. In welcher Reihenfolge die Spieler diese Aufgaben angehen, ist ihnen weitestgehend freigestellt. Sie betreten somit den nächsten Interaktionsraum. Ist diese Aufgabe bewältigt, erreichen die Spieler den nächsten Schlüsselmoment und der Prolog ist abgeschlossen.

DIE GESCHICHTE

Henry lebt alleine, von seiner Frau getrennt, in einer unbekannten Großstadt. Er bewohnt ein Apartment in einem riesigen Mehrfamilienhaus. Vor 8 Jahren ist seine Tochter bei einem Ausflug mit seinem besten Freund verschwunden. Niemand weiß, ob Sie noch lebt. Henry hat zwar die Hoffnung noch nicht aufgegeben, die Suche nach ihr jedoch schon längst. Zum Jahrestag ihres Verschwindens führt er eine Art Ritual durch und besucht ihr symbolisches Grab, eine Blume, die er am Stadtrand gepflanzt hat. Ihre sterblichen Überreste wurden nie gefunden.

Als er an diesem Tag von seinem Besuch an ihrem Grab zurückkommt, findet er in seinem Apartment seinen Freund mit durchtrennter Kehle vor. Aus Angst die Polizei könnte ihn für den Mord verdächtigen, da sein Freund für das verschwinden seiner Tochter verantwortlich war, versteckt er sich bei einer Nachbarin. Diese ist im Besitz eines Babys, was Henry sehr verwundert, da sie nie schwanger war. Sie erzählt ihm, dass es sein Kind sei.
Auf Nachfragen erzählt sie ihm, dass ein Mann das Kind vor seiner Türe abgelegt habe, während er weg war. Sie habe es daraufhin bei sich aufgenommen. Als die Polizei in die Wohnung eindringt, muss Henry fliehen. Auf der Flucht vor der Polizei und den unbekannten Mördern seines Freundes entdeckt er mehr Hinweise auf die Hintergründe der Tat.

Sein Freund, der einen Gemischtwarenladen in der Stadt betrieb, wurde von Menschenhändlern bedroht. Um sein eigenes Leben zu schützen, hat er damals Henrys Tochter an Sie verkauft und die Geschichte um Ihr Verschwinden erfunden. Seit dem wird er von Schuldgefühlen geplagt, steht er doch auch noch stets in Kontakt mit den unbekannten Menschenhändlern. Als Henrys Tochter schwanger wird und ein Kind bekommt, stiehlt er dieses, und bringt es als „Wiedergutmachung“ zu Henry. Henry schafft es, seine Tochter ausfindig zu machen und sie zu befreien.

1. Interaktive Szene: Wohnung/Prolog

Rätsel: *Das Ritual*

Zettel mit Anweisungen finden (Küche):

1. Reinige die Tränen des Himmels

⇢ Finde Glas ⇢ öffne Fenster ⇢ benutz Glas mit Regenrinne ⇢ Benutze Wasser mit Entkalker

2. Äschere den Faden des Lebens ein

⇢ öffne Schrank ⇢ nimm Faden von Kleid ⇢ benutze Feuerzeug mit Faden ⇢ benutze Asche mit Glas

3. Opfere den Staub der letzten Überreste

⇢ öffne Kommode ⇢ nimm Milchzähne ⇢ benutze Feile mit Zahn ⇢ benutze Staub mit Glas

4. Mische es mit dem Elexier das neues Leben verspricht

⇢ Finde Flüssigdünger ⇢ Mische alles

Verlasse Wohnung ⇢ Ende Prolog

PLANUNG DER INTERAKTION

Neben der Erstellung einer spannenden Geschichte ist die Planung der Interaktionsräume die eigentliche Aufgabe des Gamedesigns. Bei Spielen, die auf eine konkrete Handlung verzichten, ist Sie sogar die zentrale Aufgabe.

Im Falle dieses Point & Click Adventures war es notwendig, die Verteilung der Gegenstände und die Interaktionen mit ihnen festzulegen. Dafür wurde eine schematische Übersicht der Rätsel im Prolog erstellt. Dabei war die Gestaltung des Schwierigkeitsgrades wichtig. Da wir uns beim Prolog, also am Anfang des Spiels befinden, sollte den Spielern ein leichter Einstieg ermöglicht werden. Im späteren Spielverlauf wird der Schwierigkeitsgrad ansteigen, um den mittlerweile geübten Spielern eine stetig wachsende Herausforderung zu bieten und den Spielverlauf somit interessant und anspruchsvoll zu gestalten.

Die Variation der Schwierigkeitsgrades lag in der Anzahl der aufnehmbaren Gegenstände, ihrer Platzierung und der Offensichtlichkeit der Interaktionsmöglichkeiten. Der Zettel, den die Spieler zu Beginn finden, gibt ihnen eine gewisse Orientierung, was von ihnen zur Lösung des ersten Kapitels erwartet wird. Er senkt somit den Schwierigkeitsgrad. In späteren Kapiteln werden die Spieler selbst herausfinden müssen, welche Aufgaben zu erfüllen sind - der Schwierigkeitsgrad steigt an. Die Platzierung der Gegenstände im ersten Level ist zudem recht offensichtlich, so dass die Spieler nicht lange nach den entsprechenden Objekten suchen müssen.

MOODBOARDING – ENTWICKLUNG DER STILISTIK

Bevor die Entwicklung der Grafiken begann, wurde der Stil mit Hilfe von Moodboards entwickelt. Als Moodboards bezeichnet man Sammlungen von Fotos, Bildern und Grafiken, die in ihrer Stilistik der gewünschten Optik im Spiel entsprechen. Dieser Schritt ist vorab besonders wichtig, da man dadurch seine eigenen Ideen konkreter ausdrücken kann. Zudem können in dieser Phase stilistische Elemente der Vorlagen analysiert und für das eigene Projekt weiterentwickelt werden. In größeren Spieleproduktionen wird dieser Schritt durch eine ausgedehnte *Concept-Art-Phase* erweitert. Dabei wird ein umfangreiches Repertoire an Zeichnungen erstellt, die den Stil des Spiels visualisieren sollen.

Aufgrund der begrenzten Zeit wurde dieser Schritt hier stark eingekürzt und nur bei der Entwicklung des Hauptcharakters eingesetzt. Als Referenzen für die Optik und die Einrichtung der Räume dienten insbesondere Bilder aus den Filmen von David Lynch, allen voran *„Blue Velvet, David Lynch, 1986“*. Die surreale Einrichtung der Sets sollte sich auch in den Settings des Spiels widerspiegeln. Weitere Inspirationsquellen waren der Film *„The Machinist, Brad Anderson, 2004“*, der Animationsfilm *„Waltz with Bashir, Ari Folman, 2008“* sowie das für PC und Playstation 2 veröffentlichte Spiel *„Silent Hill 4 - The Room, Konami , 2004“*.

Für den gemalten Stil der Grafiken und insbesondere der Lichtsetzung der Settings dienten die Werke des amerikanischen Malers *Edward Hopper* als Vorlage. Seine Arbeiten zeichnen sich durch harte Lichtkontraste und eine kühle Farbgebung in Verbindung mit einer sehr geometrischen Formsprache aus. Seine Bilder können dahingehend gedeutet werden, dass sie Themen wie Einsamkeit und Isolation visualisieren. Ein Motiv, dass auch im Spiel durchgängig aufgegriffen wird.

Inland Empire
Regie: David Lynch
USA 2006

Waltz with Bashir
Regie: Ari Folman
Israel 2008

The Machinist
Regie: Brad Anderson
Spanien 2004

Blue Velvet
Regie: David Lynch
USA 1986

Silent Hill 4 - The Room
Konami
Japan 2004

Nighthawks
Edward Hopper
Öl auf Leinwand
1942

SETTINGS

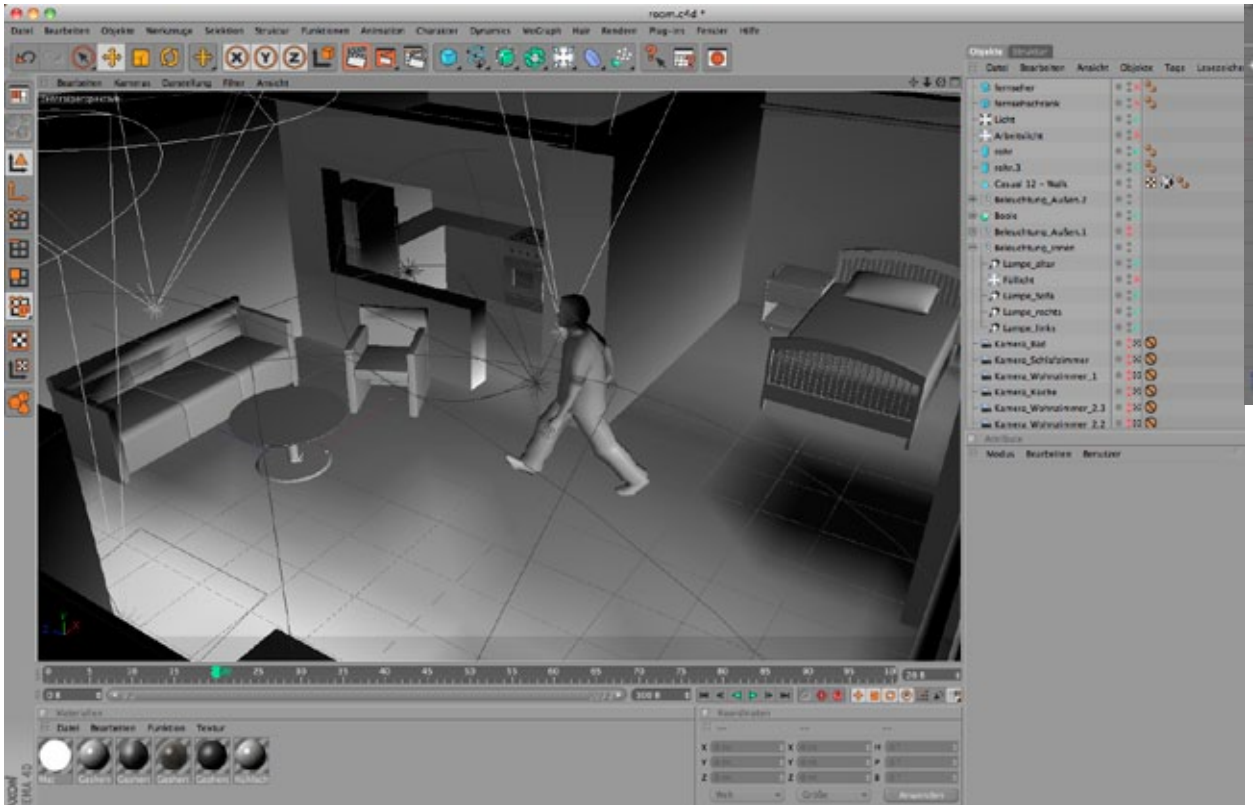
LAYOUT DER WOHNUNG

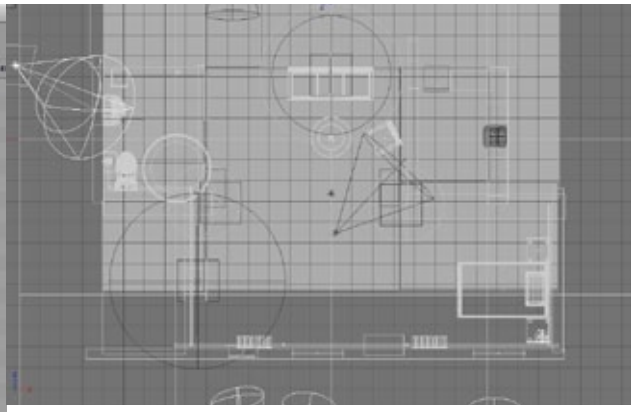
Anders als in bekannten 3rd-Person 3D-Spielen, wo die Levels in Echtzeit berechnet werden und somit die Kamera in der Regel durch den Spieler frei kontrollierbar ist, werden in 2D-Adventures vorberechnete oder illustrierte Hintergründe verwendet. Dabei wird der sichtbare Bildausschnitt schon vorab festgelegt und kann durch die Spieler nicht mehr verändert werden. Die Spielfigur bewegt sich dabei, trotz möglicher 3-dimensionaler Anmutung, nur auf einer 2-dimensionalen Fläche. Eine entsprechende Tiefenwirkung wird nur durch Skalierung der Charaktere auf der y-Achse, also der Vertikalen, erzielt. Was zunächst wie ein Nachteil gegenüber Echtzeit-3D-Spielen erscheint, erschließt jedoch für den Entwickler neue dramaturgische Möglichkeiten. Da der Spieler selbst nicht in die Kameraführung eingreifen kann, kann der Entwickler genau bestimmen, welche Informationen der Spieler zu welcher Zeit erhält. Klassische Survival-Horror-Spiele, wie zum Beispiel *„Resident Evil, Capcom, 1996“*, nutzten diese Möglichkeit um Spannung aufzubauen. Gegner wurden schon von weitem durch Geräusche angekündigt, ihre Position aber durch die feststehende Kameraperspektive verdeckten. Der Spieler wurde somit über ihren Aufenthalt im Unklaren gelassen und entsprechende Schockmomente waren vorprogrammiert.

Ein weiterer nicht zu unterschätzender Vorteil ist die einfachere Produktion. Anders als in 3D Spielen muss das Level nicht bis ins kleinste Detail aufgebaut werden - man kann sich bei der Entwicklung auf die spielrelevanten Kameraauschnitte konzentrieren. Da die Grafiken vor dem Spielstart geladen werden und somit die Berechnung der Levels in Echtzeit entfällt, wird trotz detaillierter Optik relativ wenig Prozessor- und Grafikleistung gefordert.

Raumplanung in 3D

Auf Basis der ersten Planung wurden die Layouts für die Räume erstellt. Dabei wurde zunächst die gesamte Wohnung grob in Cinema 4D aufgebaut. Diese Art der Vorplanung bringt entscheidende Vorteile. Bevor die Räume final gezeichnet wurden, konnte so leicht mit der Anordnung der einzelnen Elemente gespielt werden, und der Aufbau des Levels hinsichtlich Storytelling und Rätselstruktur optimiert werden. Zudem entfiel die arbeitsintensive Konstruktion der richtigen Perspektive beim anschließenden Zeichnen. Eine menschliche Figur als Platzhalter für den Charakter diente dazu, die Räume und Objekte in den richtigen Größenverhältnissen anzulegen. Da das Spiel ein realistisches Setting bieten sollte, war dies besonders wichtig, um die Glaubwürdigkeit der Levels sicherzustellen. Ein zu kleiner Charakter vor einem riesigen Bett oder Tisch kreiert einen „cartoonartigen“ Eindruck, der hier nicht gewünscht war. Ein weiterer Vorteil der Raumplanung in 3D war die Möglichkeit, die Szenerien vorab ausleuchten zu lassen und die Beleuch-



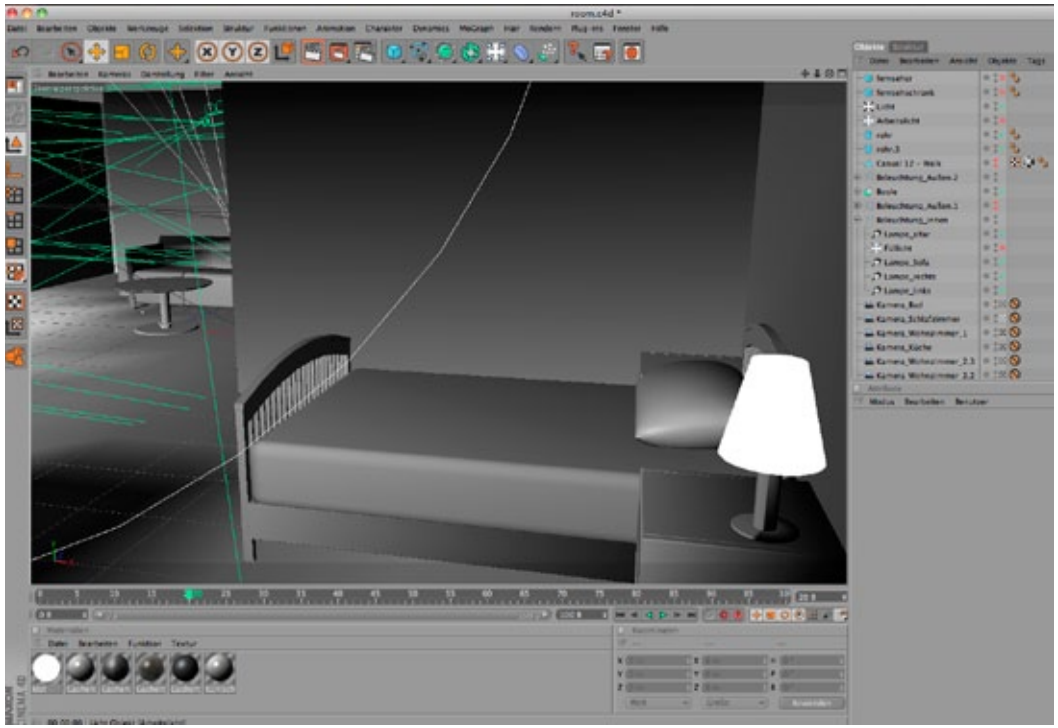


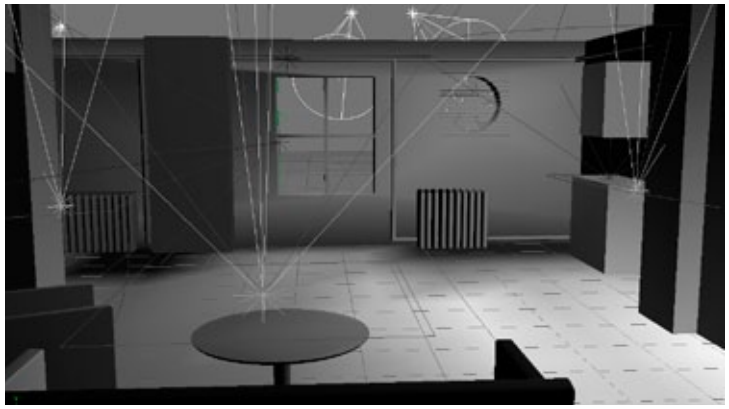
Grundriss der Wohnung

Editor Ansicht der Wohnung
mit Referenzcharakter

tung anschließend auf die Zeichnungen zu übertragen. Ähnlich wie im Film erfüllt die Lichtsetzung in Spielen dramaturgische Effekte. Zusätzlich kann sie aber auch sinnvoll in das Gameplay eingebunden werden. So kann der Spieler, bewusst oder unbewusst, durch die Beleuchtung der Levels gelenkt werden. Spiel entscheidende Gegenstände können so hervorgehoben werden oder auch, um die Schwierigkeit zu erhöhen, in dunkleren Abschnitten platziert werden.

Nachdem das erste Set komplett eingerichtet war, wurden die Kameras erzeugt welche die finalen Ansichten im Spiel repräsentieren. Dabei war es besonders wichtig auf das gewünschte Seitenverhältnis des Spiels zu achten. Da das Spiel in Full HD erstellt werden sollte, wurde ein Seitenverhältnis von 16:9 verwendet. So konnte sichergestellt werden, dass sich alle relevanten Gegenstände innerhalb des Spielbildschirms befinden und die Bildkomposition ästhetisch ansprechend ist.



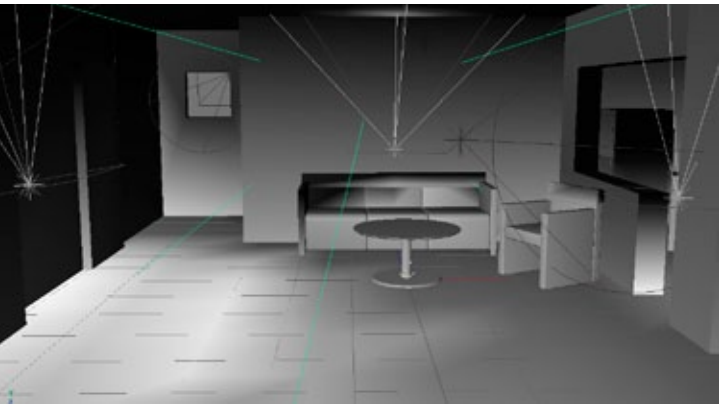


Wohnzimmer 1
Cinema 4D Editor Kamera

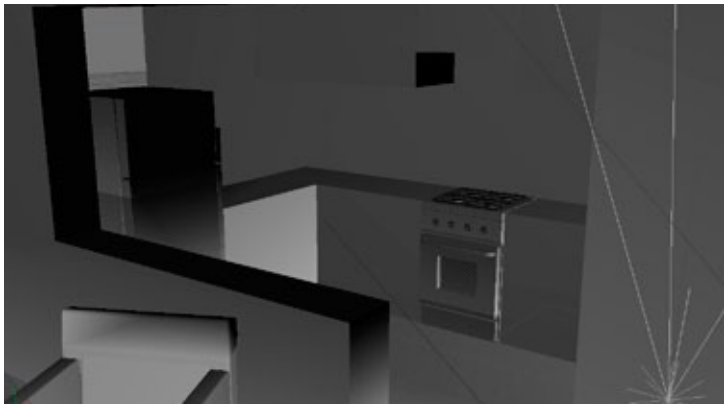
Schlafzimmer
Cinema 4D
Editor Kamera

Schlafzimmer
gerenderte Ansicht

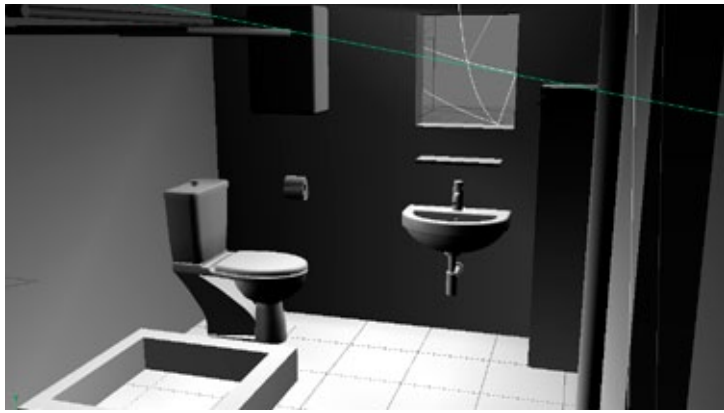
Wohnzimmer 1
gerenderte Ansicht



Wohnzimmer 2
Cinema 4D Editor Kamera



Küche
Cinema 4D Editor Kamera



Bad
Cinema 4D
Editor Kamera

Wohnzimmer 2
gerenderte Ansicht

Küche
gerenderte Ansicht

Bad
gerenderte Ansicht

ZEICHNEN DER SETS & KOLORIERUNG

Nachdem das Layout der Räume abgeschlossen war, wurden die benötigten Kameraeinstellungen gerendert. Als Rendern bezeichnet man dabei in einem 3D Programm den Arbeitsschritt, in dem das fertige Bild berechnet wird. Da, wie zuvor beschrieben, das Rendering der Levels nicht in Echtzeit während des Spiels sondern vorab, stattfindet spricht man auch von vorgerenderten Grafiken.

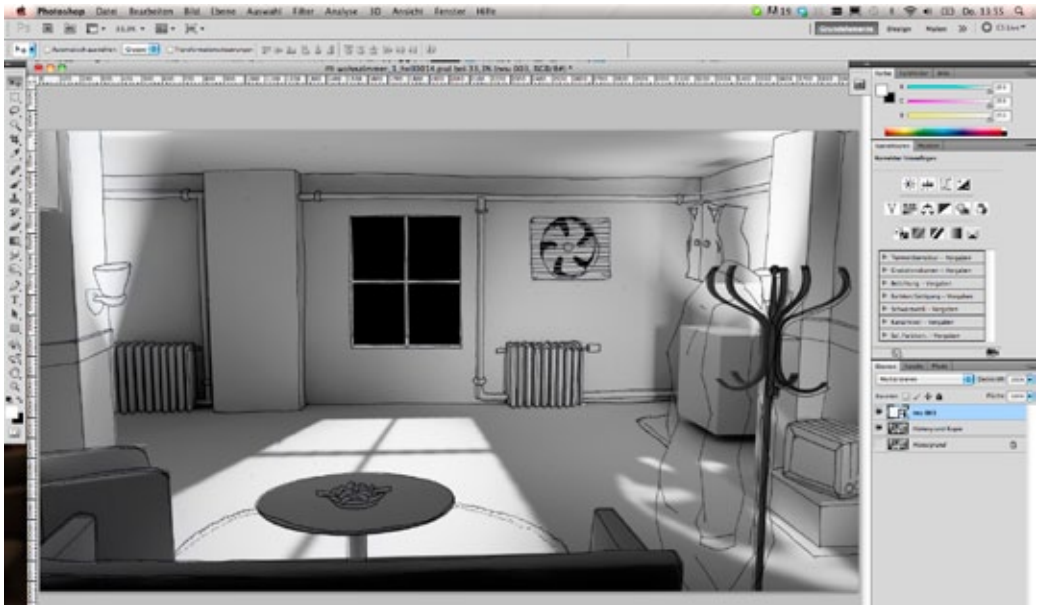
Die Renderings wurden ausgedruckt und per Hand überzeichnet. Grob angelegte Objekte wurden dabei mit Details versehen. Diesen Arbeitsschritt hätte man auch auslassen können und die Renderings direkt digital mit einem Zeichentablett am Computer kolorieren können. In diesem Projekt diente der Schritt jedoch dazu, den Grafiken einen handgezeichneten bzw. gemalten Charakter zu verleihen, der bei reinen Computerillustrationen aufgrund ihrer glatten Optik meist verloren geht. Die fertigen Zeichnungen wurden anschließend wieder eingescannt und in Photoshop koloriert.

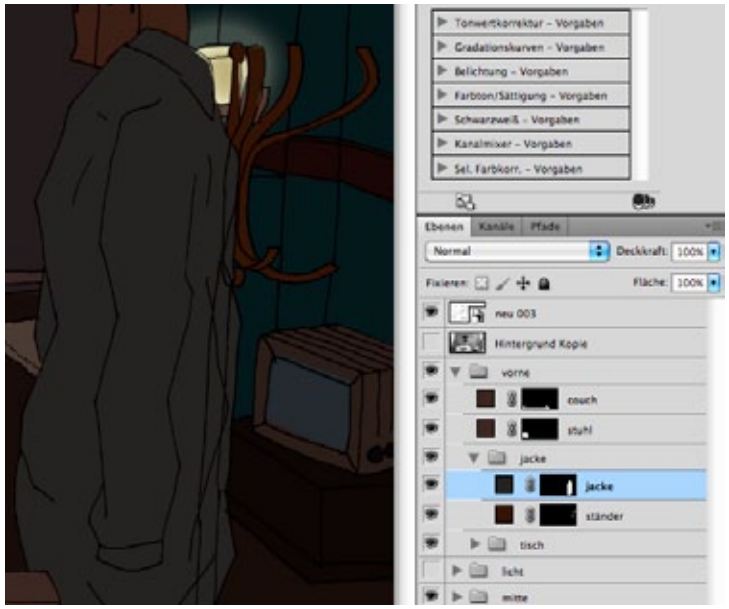
Generell ist bei der Kolorierung in Photoshop zu beachten, dass man möglichst alle Einstellungsmöglichkeiten bis zum finalen Bild beibehält. So sollten man davon absehen, das Bild auf einer einzigen Ebene zu kolorieren oder Helligkeitseinstellung, wie. z. B. die Gradiationskurven sofort anzuwenden. Sollten im späteren Verlauf Anpassungen von Farbtönen, Helligkeit oder Sättigung notwendig sein, können diese sehr kompliziert bis unmöglich werden. Es empfiehlt sich also, pro Objekt eine einzelne Farbebene mit den entsprechenden Einstellungsebenen anzulegen. Für die Platzierung der Objekte im Spiel ist es zudem notwendig, ein Objekt separat mit einem transparenten Hintergrund zu haben, zum Beispiel wenn die Spielfigur später hinter diesem Objekt stehen soll. Hat man die Grafiken der Objekte auf einer einzelnen Ebene, kann man sich später Arbeiten wie das Freistellen sparen.

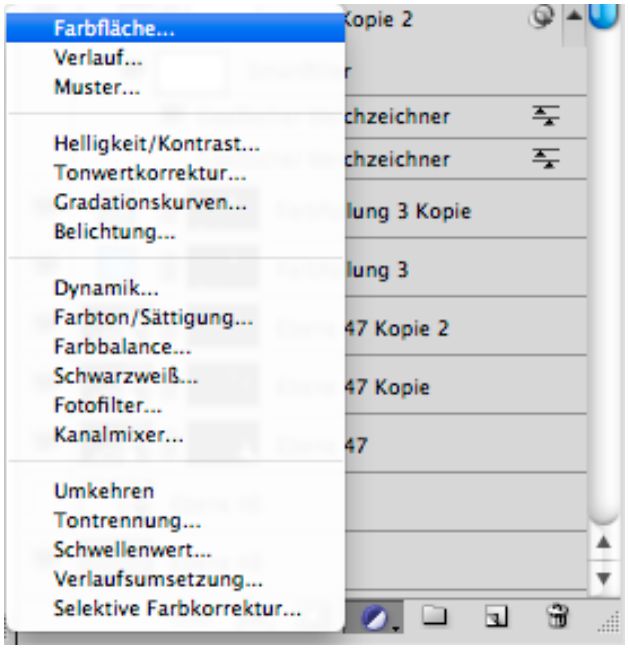
Grundsätzlich lassen sich 4 Arbeitsschritte festlegen:

1. Farbflächen erstellen
2. Schattierungen/Lichter
3. Texturierung
4. Finishing

Der Arbeitsablauf wird auf den nachfolgenden Seiten erläutert.







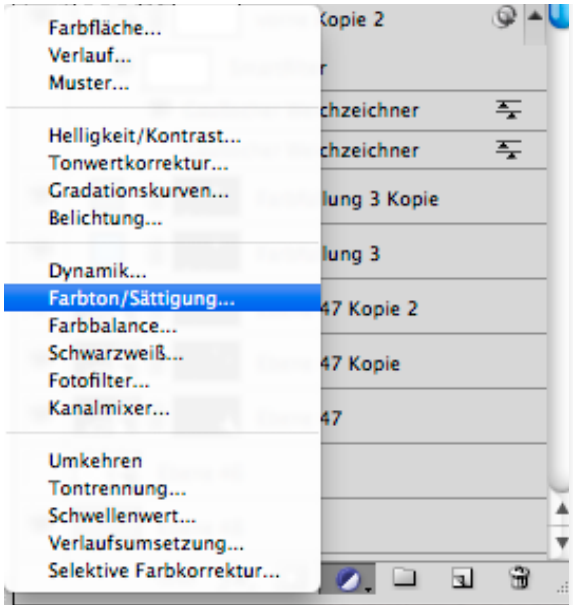
1. Farbflächen erstellen

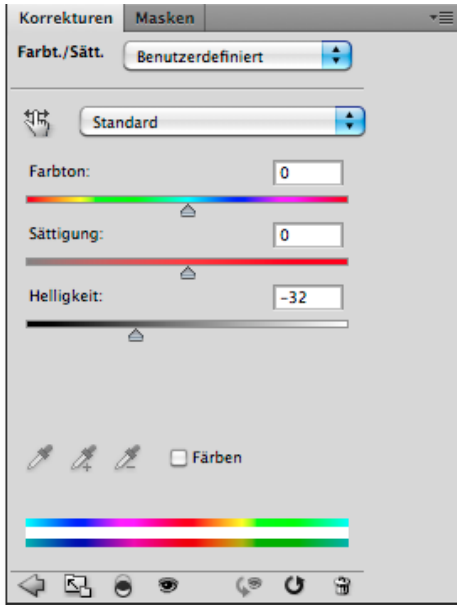
Bei diesem Arbeitsschritt werden alle Objekte auf eine farbige Fläche, die der Grundfarbe des Gegenstandes entspricht, gemalt. Als Technik hat sich dabei das Arbeiten mit editierbaren, maskierten Füllflächen bewährt. Jedes Objekt wird dabei auf einer einzelnen Füllfläche angelegt. Als Orientierung für die Konturen der Objekte dient die eingescannte Zeichnung.

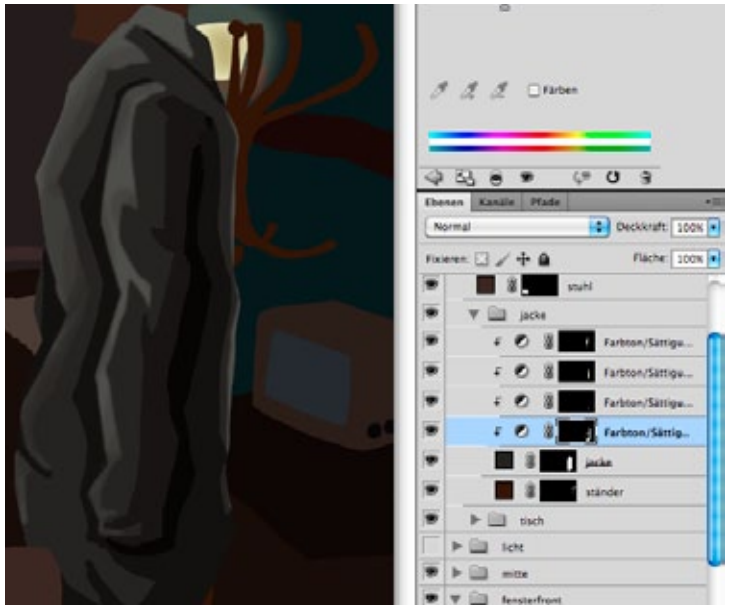
Schattierung/Lichter Setzen

Der Schattierungsprozess ist mit der entscheidenste für die Wirkung des Bildes. Je nach gewünschtem Stil können dabei sehr detaillierte, weiche Schattierungen verwendet werden, welche einen realistischen Look erzeugen. Harte Schattierungen erzeugen einen eher cartoonhaften oder abstrakten Look. In diesem Projekt sollte eine möglichst realistische Optik erzeugt werden, so dass sehr aufwändige Schattierungen benötigt wurden.

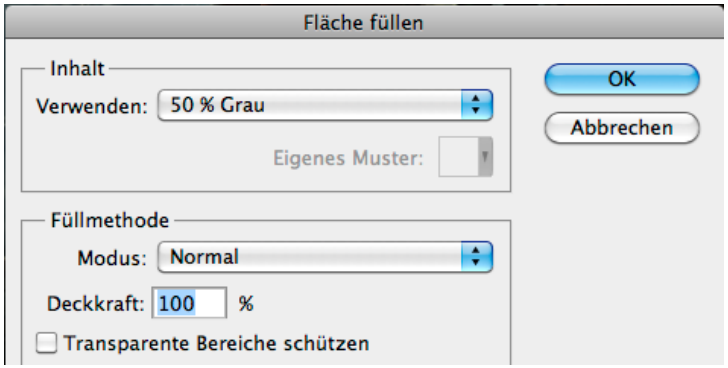
Der Schattierungsprozess lässt sich noch einmal in zwei Stufen unterteilen: dem Erstellen der *primären Schatten* und der *sekundären Schatten*. Primäre Schatten sind dabei solche, die durch den direkten Lichteinfall entstehen, zum Beispiel der Schatten eines Stuhls, der durch eine Lampe angestrahlt wird. Sekundäre Schattierungen sind solche, die indirekt durch den Lichteinfall entstehen, zum Beispiel die Schattenkante unter einem Schrank.

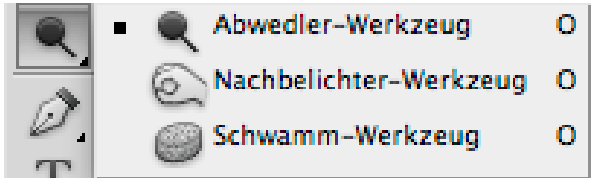


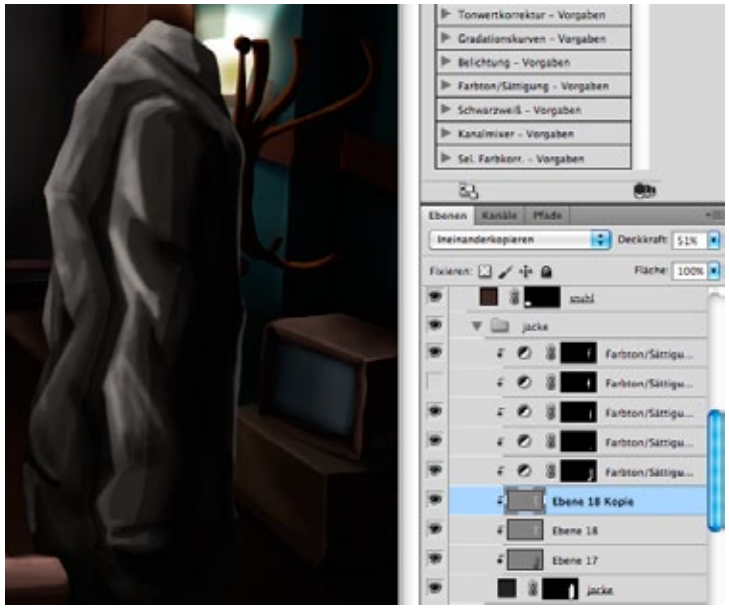


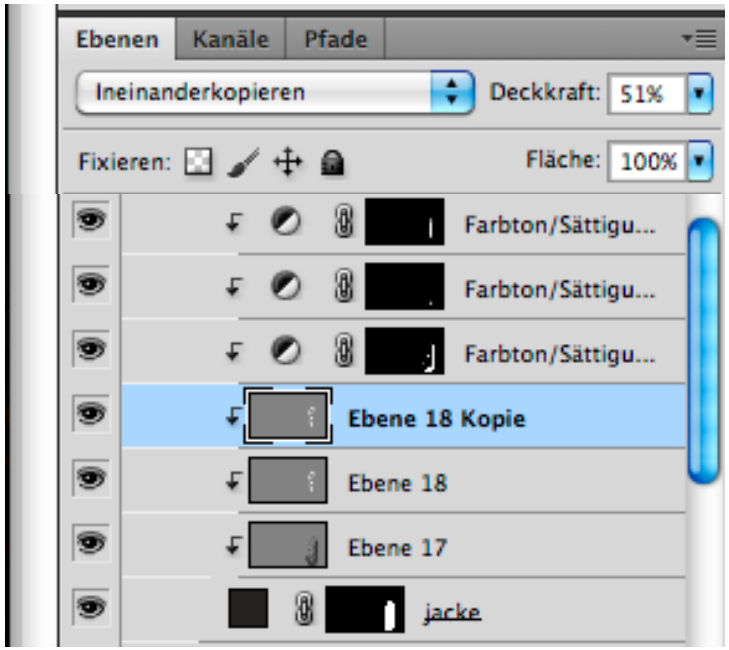


Für die Schattierung wurden 2 Techniken verwendet: die Helligkeitsänderungen über die *Einstellungsebene „Farbton/Sättigung“* sowie die *„Abwedeln und Nachbelichten“-Technik*. Über die Einstellungsebene „Farbton/Sättigung“ kann der Helligkeitswert, die Sättigung sowie der Farbton einer Ebene bestimmt werden. Die erstelle Einstellungsebene wird mit einer Maske versehen auf denen die eigentliche Schattierung gemalt wird. Dies eignet sich besonders für großflächige Schattierungen, zum Beispiel die unterschiedliche Helligkeit der von Wand, Decke und Boden in einem Raum. Empfehlenswert ist es, die Einstellungsebenen der darunterliegenden Farbfläche mit den Konturen des Objekts zuzuordnen. Dies erreicht man durch Gedrückthalten der *„Alt-Taste“* und einem Mausklick zwischen beiden Ebenen. Dadurch bezieht sich die Einstellungsebene ausschließlich auf die darunterliegende Füllebene. Man kann also auch über die Ränder malen, ohne ein anderes Objekt mit einzufärben. Es lasse sich beliebig viele Ebenen gemeinsam einer darunterliegenden Ebene zuordnen. Auf diese Art und Weise wurden die primären Schattierungen der Objekte erstellt.



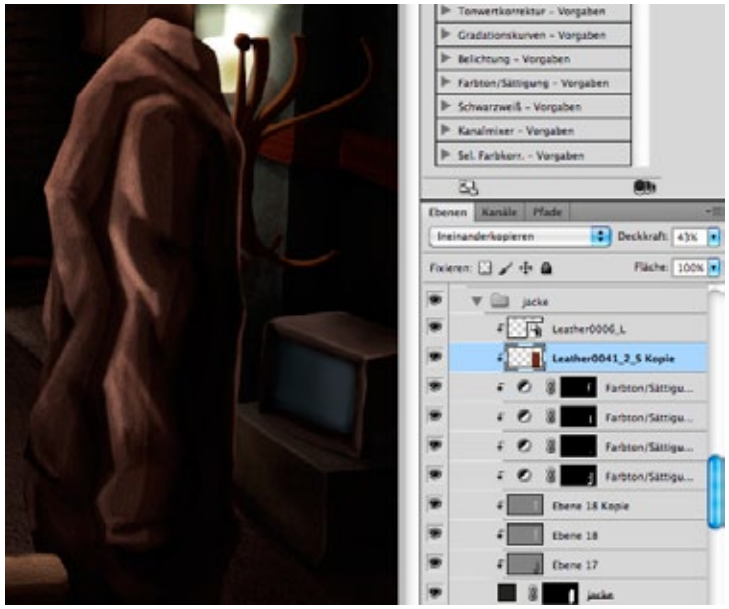






Für die sekundären Schatten kam die Technik *„Abwedeln und Nachbelichten“* zum Einsatz. Dabei erstellt man eine Ebene gefüllt mit einer 50%-igen Graufläche und stellt den Ebenenmodus auf „Ineinanderkopieren“, wodurch die Graufläche zunächst nicht sichtbar ist. Nun kann man mit den Werkzeugen *„Farbig Nachbelichten“* und *„Farbig Abwedeln“* die weichen Schattierungen im Objekt erzeugen.

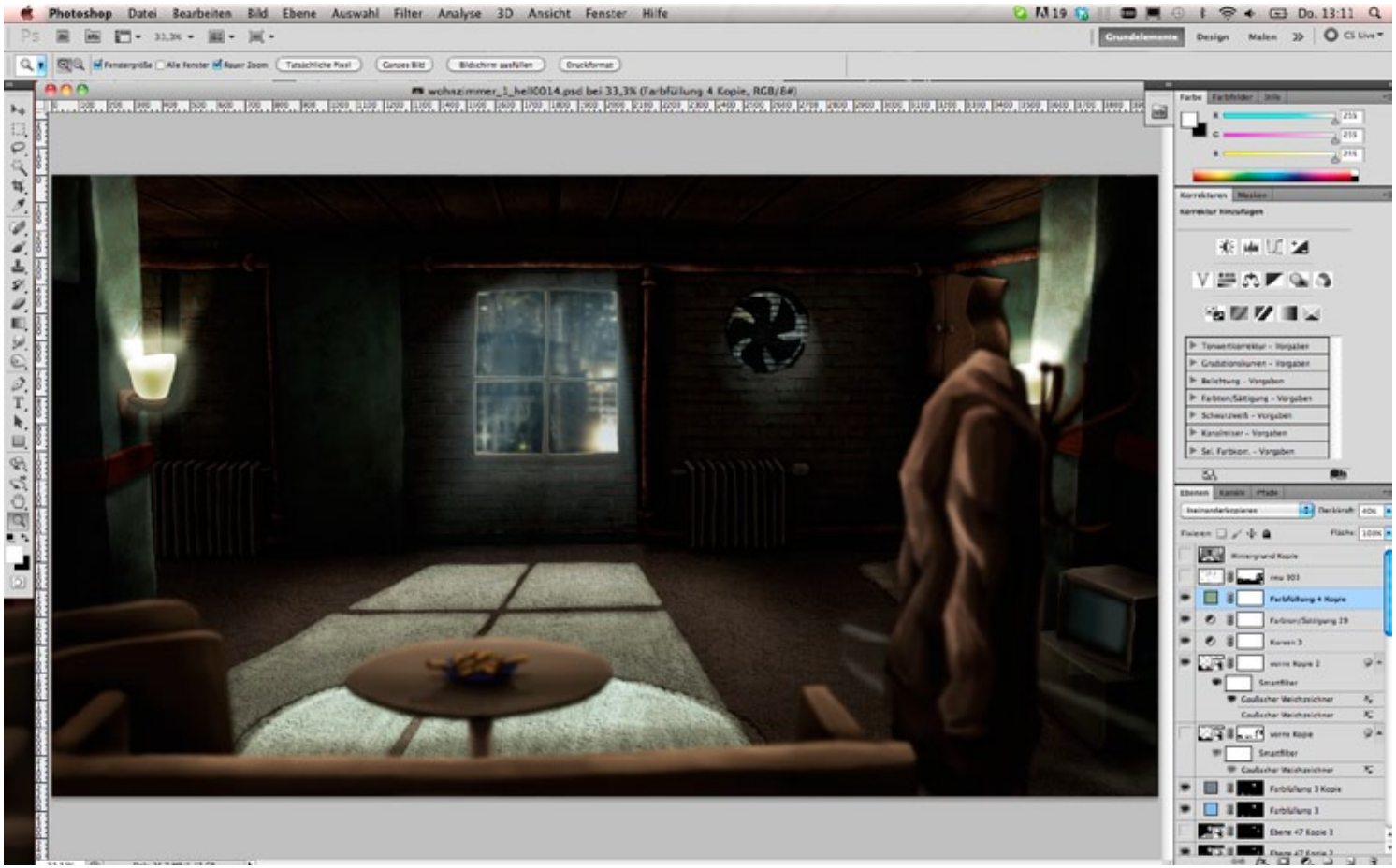
Leder Textur



Texturierung

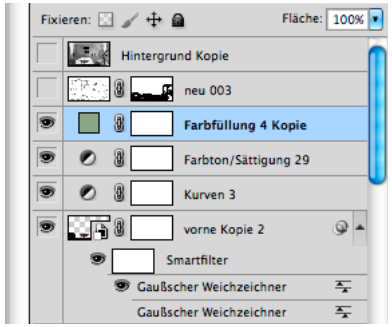
Texturen geben den Oberflächen der Objekte zusätzlich Charakter und Struktur. Texturen von Leder, Holz, Stein etc. lassen sich als Bilder in diversen Datenbanken im Internet, zum Beispiel *www.cgtextures.com*, finden und kostenlos herunterladen. Anschließend werden die Bilder als gesonderte Ebene in Photoshop geladen und einer darunterliegenden Ebene per *„Alt-Taste“* zugeordnet. Durch die Verwendung von Ebenen-Modi wie „Ineinanderkopieren“ oder „Multiplizieren“ werden die Textur-Ebenen mit der darunterliegenden Farbfläche verrechnet - die Farbfläche erhält die entsprechende Struktur.

Generell lässt sich sagen, das die Schritte „Schattierung“ und „Texturierung“ sich in ihrer Reihenfolge auch abwechseln können. Manchmal ist es sinnvoll, einem Objekt zuerst eine Textur zuzuordnen um die Schattenwirkung des texturierten Objekts besser bestimmen zu können.



Finishing

In dem letzten Schritt werden über Einstellungsebenen wie zum Beispiel die Gradiationskurven der Gesamteindruck des Bildes wie gewünscht angepasst.



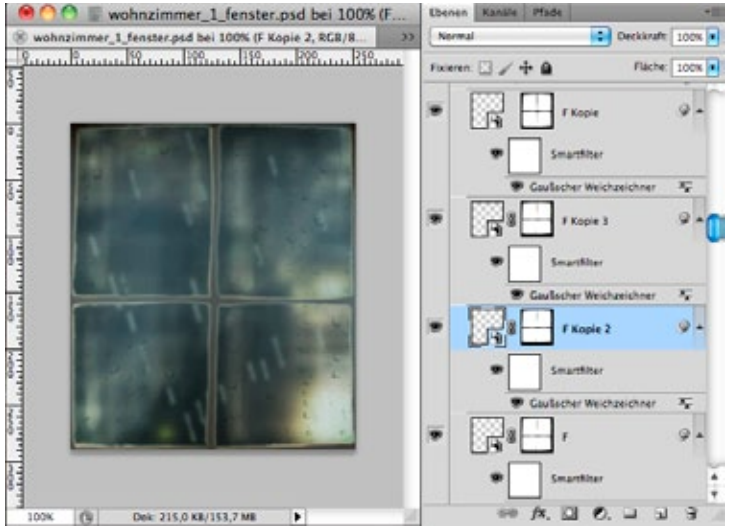
ANIMATION DER SETS

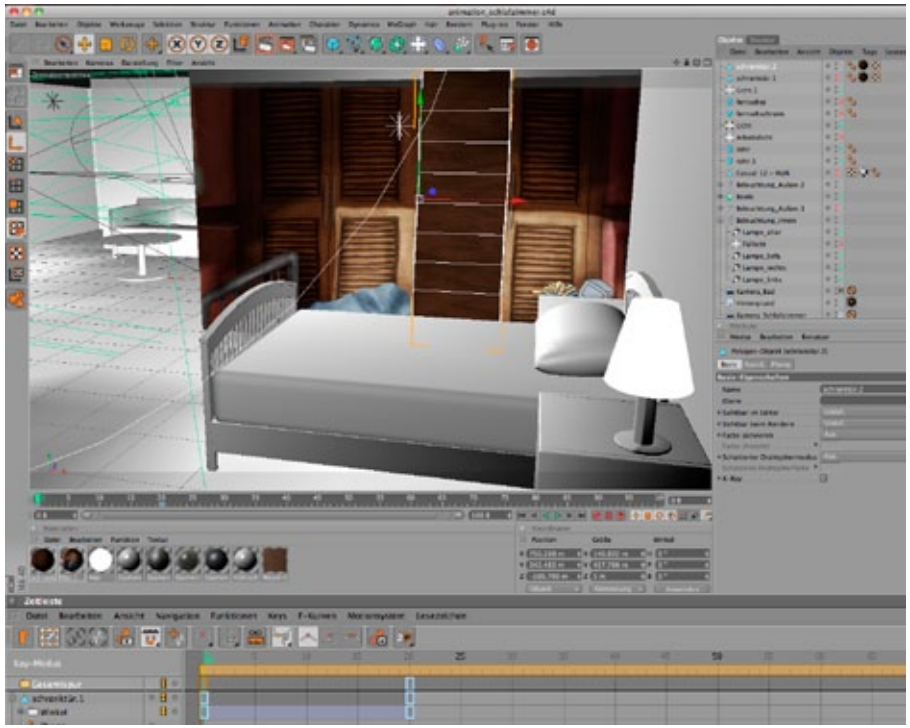
Um den Hintergründen im Spiel mehr Leben einzuhauchen, wurden bewegliche Objekte wie Regen oder Türen animiert. Dabei wurden zwei verschiedene Vorgehensweisen gewählt: eine für 2-dimensionale Objekte, welche ihre Perspektive während der Animation nicht ändern, sowie eine für 3-dimensionale Objekte wie aufschwingende Türen, welche während der Animation ihre Perspektive ändern.

Video-Ausgangsmaterial
Regen

Animation von 2-dimensionalen Objekten

Der Regen im Wohnzimmer wurde mit Hilfe der *„Videoebenen“* in Photoshop animiert. Ähnlich wie reine Bilddateien ist es in Photoshop auch möglich, Bildsequenzen und Videos auf einer Ebene zu platzieren und zu bearbeiten. Um Zeit zu sparen wurde für den Regen auf bestehendes Videomaterial zurückgegriffen. Die Bildsequenz wurde über die Funktion *„Ebene ⇢ Videoebene ⇢ neue Videoebene aus Datei“* in Photoshop importiert und mit Hilfe der Gradiationskurven und Maskierungen an die Szene angepasst. Um unerwünschte Zeitunterschiede und fehlerhafte Animationen zu vermeiden war darauf zu achten, dass die Framerate im Dokument den Vorgaben des eingeladenen Videos, in diesem Fall 25 fps, entspricht.
Ein Vorteil die Videos direkt in Photoshop zu bearbeiten waren die optimalen Einstellungsmöglichkeiten um das Video an die gesamte Szene anzupassen.



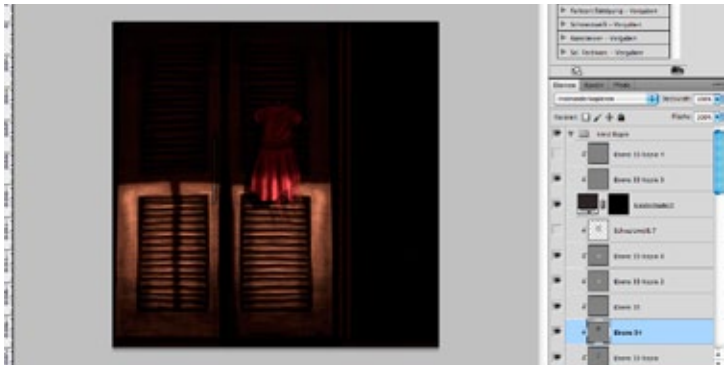


Animation von 3-dimensionalen Objekten

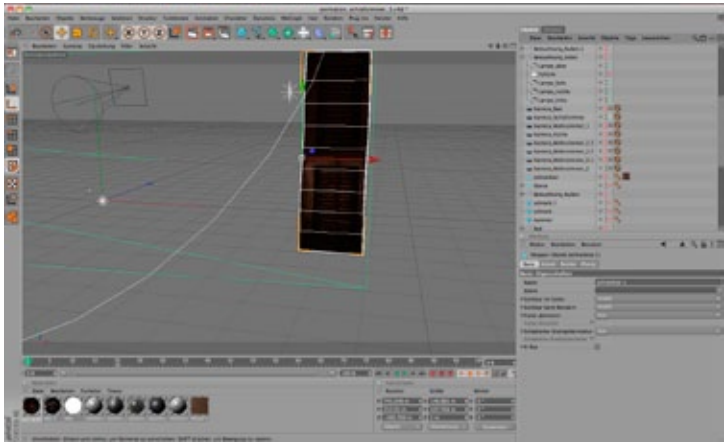
Bei Objekten die während der Animation ihre Perspektive wechseln war eine andere Vorgehensweise erforderlich. Da sich die Perspektive des Objekts in jedem Frame ändert, hätte man für jede Animationsstufe des Objekts *Einzelbilder* zeichnen müssen. Aufgrund des hohen Detailgrades der Objekte wäre dies bei einer normalen *Frame-by-Frame Animation* aus Zeitgründen unmöglich gewesen, deshalb wurde für diese Animationen der Weg über *Cinema 4D* gewählt.
Da die Räume und Kameraeinstellungen bereits in 3D vorlagen, wurde für die zu animierenden Objekte ein weiteres 3D-Objekt, im Falle der Türe ein flaches Rechteck eingefügt. Das Aufschwingen der Türe konnte nun mithilfe der *Keyframe-Animation* erstellt werden. Nun musste dem Objekt noch die entsprechende Textur des Szenenobjekts zugewiesen werden, um es nahtlos in die Szene zu integrieren. Dafür wurde in Cinema 4D über die Funktion *„Objekt backen“* eine *UV-Map* für das Objekt erstellt. Eine UV-Map beschreibt, wie ein flaches Texturbild um ein 3-dimensionales Objekt gewickelt wird und gewährleistet die korrekte Darstellung der Textur.
Das Ergebnis der Funktion „Objekt backen“ ist ein neues 3D-Objekt mit den entsprechenden UV-Koordinaten sowie ein 2-dimensionales Abbild der Textur. Dieses Texturbild wurde nun in Photoshop importiert und bearbeitet. Dabei wurden ihm die Texturen des Objekts aus der fertigen Szene zugewiesen.

UV-Map
in Photoshop

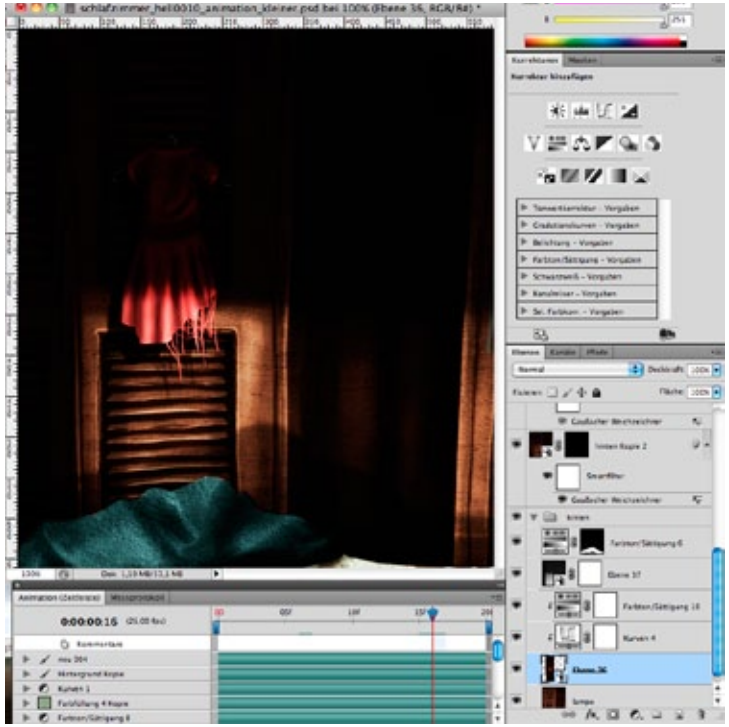
Fertig texturierte UV-Map
in Photoshop



Mit UV-Map texturiertes Objekt
in Cinema 4D



Die fertige Texturmap wurde anschließend wieder in das UV-Material in Cinema 4D importiert und so dem Objekt zugewiesen. Dabei war zu beachten, dass die neue Textur in den „Leuchten“-Kanal des Materials importiert wird, da die Farbdarstellung ansonsten nicht dem Objekt der Originalszene entsprechen würde.
Anschließend wurden die Einzelbilder des animierten Objekts gerendert. Für eine bessere Integration in die Szene wurde zusätzlich eine Darstellung der Schatten in der Szene, eine *„Shadowmap“*, gerendert.
Die entstandene Einzelbildsequenz wurde in Photoshop importiert und mit einer entsprechenden Mask in die Szene integriert.



ERSTELLUNG DER GRAFIKEN

Die fertigen Grafiken der Räume konnten nun für das Spiel vorbereitet werden. Vollflächige Grafiken wurden dazu aufgrund der besseren Komprimierung als *JPG-Datei*, Grafiken mit transparenten Elementen als *PNG-Grafiken* mit Alphakanal gespeichert. Für jeden Raum wurden folgende Grafiken erstellt:

1. Hintergrundbild mit der gesamten Szenerie inkl. aller nicht aufnehmbaren Objekte
2. Objektbilder als PNGs mit transparentem Hintergrund
3. Einzelbilder der Animation
4. Graustufen Lightmap zur Beleuchtung des Characters

Die Lightmap stellt in diesem Falle noch eine zusätzliche Besonderheit dar. Sie beschreibt im Spiel, wie die Spielfigur innerhalb der Szenerie beleuchtet wird. Dazu wurde in Photoshop eine vollflächige Ebene angelegt und die Positionen, in denen die Spielfigur dunkler erscheinen sollte, dunkelgrau eingefärbt, hell erleuchtete Stellen nahezu weiß. Durch die Verwendung von Farben war auch eine farbige Beleuchtung der Spielfigur möglich.

Bei den Einzelbildern der Animation war es besonders wichtig, auf eine möglichst geringe Dateigröße zu achten. Da die Einzelbilder im Spiel geladen werden müssen, kann es bei großen Dateien zu Rucklern und Aussetzern im Spiel kommen.

Hintergrundbild
JPG, 250 KB
1920 x 1080 px

Lightmap
JPG, 50 KB

Vordergrund Objekt
PNG, 800 KB

Animationen
JPG-Sequenz

CHARAKTER
ENTWICKLUNG

CHARAKTER MOODBOARDS

Eine besondere Herausforderung stellte die Entwicklung des Hauptcharakters dar. Er musste thematisch zum Spiel passen, durch sein Aussehen bereits auf bestimmte Charakterzüge schließen lassen sowie sich optisch möglichst homogen in die Szenerie einfügen.
Als erster Schritt wurde eine kurze Charakterisierung der Hauptfigur entworfen:

Protagonist
Henry, Alter: Mitte 40

„Depressiv, zynischer Humor, das Leben zieht an ihm vorbei („Jeden Tag lebe ich wie in einem Vakuum, ich arbeite, bin mit den Gedanken nie wirklich dabei. Jeden Tag hoffe ich auf eine Neues, eine wundersame Botschaft meiner Tochter zu erhalten, nur ein kleines Indiz dafür, dass sie noch am Leben ist. Alles andere ist unwichtig. Und jeden Abend macht sich dieses Gefühl in mir breit, als ob sich eine kalte Flamme durch meinen Körper frisst und ich weiß, dass es diese Hoffnung nicht gibt)

Seine Frau hat Ihn nach dem Verlust der Tochter verlassen. Er hasst sie dafür, dass sie ihn alleine gelassen hat (Auf gemeinsamen Fotos in der Wohnung ist ihr Gesicht durchgestrichen, nur er und seine Tochter sind zu sehen.)“

Das optische Erscheinungsbild des Charakters musste also diese Charakterzüge und auch die Hintergrundgeschichte der Figur visuell vermitteln. Als weitere Anhaltspunkte für die Entwicklung wurden Referenzbilder von ähnlichen Charakteren zusammengestellt.

Roger
OT: Metropia
Regie: Tarik Saleh
Schweden 2009

Ari Folman
OT: Waltz with Bashir
Regie: Ari Folman
Israel 2008

Trevor Reznik
OT: The Machinist
Regie: Brad Anderson
Spanien 2004

Charakter-Konzept
Künstler Unbekannt

Henry Spencer
OT: Eraserhead
Regie: David Lynch
USA 1977

Nico Bellic
OT: Grand Theft Auto 4
Rockstar North
USA 2008

CHARAKTER SKIZZEN

Auf Basis der Charakterisierung sowie den Referenzbildern wurden erste Skizzen des Hauptcharakters erstellt. Dabei entwickelten sich zwei unterschiedliche Richtungen, ein eher cartoonhafter Ansatz sowie eine realistische Darstellung. Beide Charakterskizzen wurden gescannt und in Photoshop digital koloriert. Anschließend wurden die Charaktere testweise in die fertigen Szenen eingefügt um ihre optische Wirkung hinsichtlich Stil und Homogenität besser beurteilen zu können.

Charakter Linie 1
mit realistischen
Attributen

Charakter Linie 2
mit cartoonhaften
Attributen

FINALISIERUNG DES CHARAKTERS

Nach einer ersten Feedbackrunde stellte sich schnell heraus, dass der realistische Charakteransatz besser zum Setting und den bereits erstellten Grafiken passen würde. Die überzogene Stilistik des Cartoon-Charakters harmoniert einfach nicht mit der realistischen Optik der Settings. Da der realistische Charakter jedoch nicht sehr ausdrucksstark und charakterlos wirkte, wurden in einem zweiten Schritt weitere realistische Charakterskizze erstellt, welche bis zum finalen Charakter ausgearbeitet wurden. Besonders wichtig war dabei die Gestaltung des Gesichts, da es sofort die Emotionen des Charakters ausdrücken sollte. Durch Elemente wie die leicht heruntergezogenen Augenwinkel, die verlängerte Nase sowie den weit unten angesetzten Mund entstand ein Gesicht, das die Depression und Niedergeschlagenheit der Hauptfigur möglichst genau ausdrückt.

Die Skizzen wurden anschließend in Photoshop nach der gängigen Methode koloriert. Besonderer Augenmerk lag dabei auf der Schattierung des Charakters. Diese durfte nicht zu spezifische, also nicht aus einem extremen Winkel erfolgen, da die Spielfigur später in verschiedenen Beleuchtungssituationen stehen würde und sich trotzdem harmonisch in die vorhandene Lichtsituation einfügen musste. Um dem Charakter trotzdem Räumlichkeit und Tiefe zu geben, wurde die Lichtsituation auf eine Beleuchtung von rechts oben festgelegt. Dieser Schritt ist wichtig, da eine nicht klar definierte Beleuchtung des Charakter zu einem unstimmigen Gesamtbild führt.

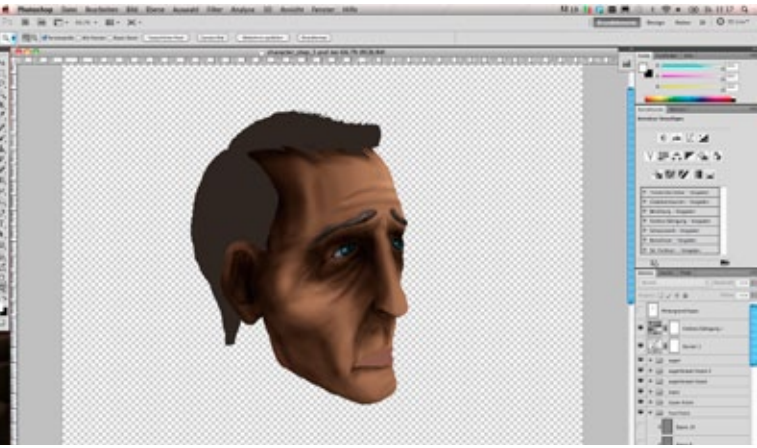
Da der Ablauf im Wesentlichen der Kolorierung der Settings gleicht, wird er an dieser Stelle nur kurz wiedergegeben.

Ein sehr gutes und detailliertes Tutorial zur Kolorierung von Charakteren in Photoshop findet sich hier:

http://psd.fanextra.com/tutorials/drawing/digital-painting-less-on-monkey-island-2-lechuck%E2%80%99s-revenge-special-edition/
Generell gibt es aus technischer Sicht verschiedene Möglich

Schritt 1
Eingescannte Zeichnung in Photoshop angelegt.

Schritt 2
Farbige Füllflächen für alle Elemente des Charakters anlegen.



Schritt 3
Schattierungen setzen. Dafür für jede Ebene eine Ebene gefüllt mit 50% Grau, Ebenenmodus „Ineinanderkopieren“ anlegen. Anschließend die Graustufenebene mit dem Werkzeug *„Nachbelichten“* an den gewünschten Stellen abdunkeln.

Schritt 4
Lichter setzen. Gleiche Technik wie bei den Schattierungen, als Werkzeug wird jedoch *„Abwedeln“* verwendet.

Schritt 5
Finishing. Über Einstellungsebenen wie *„Gradiationskurven“* und *„Farbton/Sättigung“* die Luminanz des Bildes anpassen.

CHARAKTER-ANIMATION GRUNDLAGEN

Bevor man mit der Animation eines Charakters beginnt, sollte man sich mit den Prinzipien der Animation vertraut machen. Um eine realistische und lebendig wirkende Animation zu erstellen, gibt es bestimmte Grundregeln. Diese wurden erstmals von den beiden Disney Animatoren Ollie Johnston und Frank Thomas in ihrem 1981 erschienenen Buch *„The Illusion of Life: Disney Animation“* veröffentlicht. Sie gehen auf die Entwicklung der Animation seit den frühen 1930er Jahren zurück und besitzen auch heute noch, im Zeitalter von industriell gefertigter 3D-Animationen ihre Gültigkeit.
Besonders wichtig für eine realistische Animation sind dabei die Punkte Timing, also der Bewegungsabstand der einzelnen Bildframes, sowie das Dehnen und Stauchen des Charakters gemäß der Bewegungen. Da die Bewegungen des Charakters möglichst realistisch wirken sollten wurde auf eine comichafte Überteibung der Bewegungen verzichtet.

Stauchen und Dehnen
Unter *„Stauchen und Dehnen“* versteht man die Art und Weise wie sich ein Objekt bei Bewegungen verzerrt. Als Beispiel sei ein springender Ball genannt. In der Luft behält er seine Kugelform, trifft er auf den Boden auf, verzerrt er sich zu einer Ellipse. Ebenso verformen Sie die Muskeln eines Charakters in der Bewegung.

Ausholen
Vor jeder ausgeführten Bewegung gibt es eine Phase des Ausholens. So muss eine Person, bevor Sie einen Ball wirft, zuerst mit dem Arm ausholen. Das Ausholen bereitet den Betrachter auf die bevorstehende Bewegung vor und lässt diese realistischer erscheinen.

„Straight ahead action“ und „pose to pose“
Diese beiden Begriffe beschreiben zwei unterschiedliche Herangehensweisen eine Animation zu erstellen. Bei der *„Straight ahead action“* werden alle Einzelbilder der Animation nacheinander gezeichnet. Bei der *„Pose-to-Pose“* werden zuerst die *Keyframes*, also die Extremposen der Bewegung gezeichnet. Anschließend werden die dazwischen liegenden Bilder, die *Inbetweens*, erstellt.

vgl. Ollie Johnston und Frank Thomas, The Illusion of Life: Disney Animation, Seiten 47-69

Inszenierung
Wie auch im Theater oder im Film muss in der Animation eine Bühne für den Charakter geschaffen werden. Die Bildkomposition sollte dabei so genutzt werden, dass der Blick des Betrachter zu den wichtigen Objekten gelenkt wird.

Nachfolgende Bewegung und überlappende Aktionen
Unter der nachfolgenden Bewegung versteht man, das ein Körperteil nach einer ausgeführten Bewegung nie sofort stoppt. Bleibt eine Person aus dem Gang plötzlich stehen, schwingen die Arme immer noch ein Stück weiter.
Unter überlappender Aktion versteht man, dass sich zwei Körperteile nie völlig parallel, sondern leicht zeitversetzt Bewegen. Bei Gang eines Menschen bewegen sich so z.B. die Arme leicht zeitversetzt zu den Beinen.

Ease in & Ease out
Jede Bewegung hat eine Phase des Beschleunigens und des Abbremsens, sie ist also niemals linear. Deshalb muss der visuelle Abstand zwischen den Einzelbildern einer Animation Anfangs kleiner, in der Mitte größer und vor der Extrempose wieder kleiner sein.

Achsen
Jede Bewegung folgt einer Achse, d. h. Gliedmaßen bewegen sich nie auf einer geraden Linie sondern entlang eines Bogens.

Sekundäre Bewegungen
Die Bewegungen eines Charakters wirken lebendiger, wenn andere Körperteile der Bewegung folgen. Bei einem Gang werden zum Beispiel nicht nur die Beine bewegt, die Arme schwingen stets mit.

Timing
Das Timing bestimmt die Schnelligkeit des Bewegungsablaufs. wird eine Bewegung zu langsam oder zu schnell gezeichnet wirkt sie unglaubwürdig.

Überteibung
Bei Cartoon-Animationen sollten Bewegungen immer besonders übertrieben dargestellt werden. Dadurch wirken sie lebendiger. Zu dezente Bewegungen können schnell steif und und ungelenk wirken.

Zeichenskills
Um einen glaubhaften Charakter zu erschaffen, benötigt man gute Zeichenfähigkeiten. Die zeichnerische Umsetzung eines Charakters muss aus verschiedenen Perspektiven und unter Berücksichtigung der korrekten Anatomie erfolgen.

Ausstrahlung
Ein Charakter benötigt, ähnlich eines Schauspielers, Ausstrahlung und Charisma.

TECHNISCHE UMSETZUNG DER ANIMATION

Generell gibt es aus technischer Sicht verschiedene Möglichkeiten der Charakter Animation. Zum einen die *klassische Zeichentrick Animation*. Hier werden die einzelnen Bewegungsstufen jeweils als Einzelbilder gezeichnet und anschließend in schneller Folge abgespielt. Der Charakter muss dafür in jedem Bild komplett neu gezeichnet werden, wodurch diese Animationstechnik die zeitintensivste darstellt. Es entsteht bei einer ausreichend hohen Anzahl der Einzelbilder, mindestens 12 Bilder pro Sekunde, der Eindruck einer flüssigen Bewegung.
Eine weitere traditionelle Methode ist die der *Cutout-Technik*. Dabei wird der Charakter in einzelne Elemente wie Oberarm, Arm, Körper, Kopf, Oberschenkel, Unterschenkel, Fuß zerlegt. Die einzelnen Elemente werden an ihren Gelenken von Bild zu Bild so gedreht und verschoben, dass ebenfalls eine Bildfolge entsteht, die abgespielt eine flüssige Bewegung ergibt. Diese Technik ist Artverwandt mit der *Stop-Motion Animation* und wurde besonders von Lotte Reiniger in Ihrem Film *„Die Abenteuer des Prinzen Achmed, 1926“* entwickelt und in ihren darauffolgenden Arbeiten perfektioniert.

Während zu den Anfängen noch analog ausgeschnittene Teile verwendet wurden, bieten aktuelle Animationsprogramme wie *Adobe After Effects* oder *Adobe Flash* heutzutage auch die Möglichkeit, die Cutout-Technik mit rein digitalem Bildmaterial zu erstellen. Charakteristisch für die Cutout-Technik ist der leicht kantige, technische Look der Bewegungen, der auch heutzutage noch Animationsserien wie z. B. „South Park“ ihre leicht „trashige“ Ausstrahlung verleiht. Verglichen mit der klassischen Zeichentrick Animation ist diese Technik recht zeit- und kostensparend umzusetzen, da die Charaktere nur einmal erstellt werden müssen und die Bewegungen anschließend durch Ausrichten der einzelnen Elemente erzeugt werden. Dass mit dieser Technik auch annähernd natürliche Bewegungen erzeugt werden können, zeigt der dokumentarische Animationsfilm *„Waltz with Bashir, Ari Folman, 2008“*. Trotzdem haftet den erstellten Bewegungen immer noch ein gewisser technischer Look an.

Die heutzutage verbreitetste Technik ist die der *3D-Animation*. Anders als bei den voran beschriebenen Animationstechniken liegen die Charaktere hier nicht als 2D-Objekte sondern als *3D-Polygonmodell* vor. Diese Modelle sind mit einem Skelett, dem sogenannten *RIG* ausgestattet, wodurch sich die Charaktere entsprechend realistisch Bewegen lassen.

Im Bereich des Animationsfilms wurde diese Technik besonders durch die *Pixar Studios* ab den 80er Jahren vorangetrieben und wird heutzutage in Animationsfilmen, 3D-Spielen oder für digitale Charaktere und Special-Effects in Realfilmen verwendet. Die Produktion dieser Animationen erfordert einiges an technischem Know-How und ist sehr zeitintensiv. Die Ske-

lette der Charaktere lassen sich per Hand animieren. Zusätzlich können durch *Motion Capturing* aufgezeichnete Bewegungen realer Schauspieler auf das Skelett des animierten Charakter übertragen werden.

Adaption auf den Spielcharakter

Die geeignete Animationstechnik für den Hauptcharakter des Spiel zu finden erwies sich als Herausforderung. Da die Hauptfigur nur als 2-dimensionales Objekt vorlag und das entsprechende Modelling einer derart detaillierten Figur in 3D zu zeitaufwändig war, schied die 3D Animation schon vorab aus. Gegen die klassische Zeichentrick Animation sprach, dass die Figur zu detailliert gezeichnet war. Diesen Detailgrad in jedem Einzelbild erneut zu zeichnen wäre schlichtweg zu zeitintensiv und würde bei Ungenauigkeiten in der Schattierung zwischen den einzelnen Bildern nicht gut aussehen. Bei klassischen Zeichentrick-Produktionen wird dies umgangen, indem die Charaktere sehr flächig und ohne starke Schattierungen erstellt werden. Im vorliegenden Fall sollte die Hauptfigur aber genau diesen Detailgrad aufweisen, um sich harmonisch in die Szenerie einzufügen. Die Cutout-Technik wäre demnach eine Möglichkeit, die Animation im entsprechenden Zeitrahmen umzusetzen. Jedoch war der sehr technische Look dieser Animationstechnik für die fertige Spielfigur nicht gewünscht, wodurch diese Technik nur in Grundzügen verwendet wurde und durch eine weitere Technik ergänzt wurde, die der *Marionetten-Animation in After Effects*. Hierbei lassen sich einer 2-dimensionalen Grafik Gelenkpunkte zuordnen, wobei das Programm der Grafik automatisch ein Gitternetz zuweist wodurch sie verzerrt werden kann. Im Gegensatz zur Cutout-Animation wirken die dadurch erstellten Bewegungen aufgrund der Verzerrung und Stauchung der Grafiken deutlich organischer, weshalb diese Technik auch für die Animation des Hauptcharakters verwendet wurde.

ARBEITSSCHRITTE DER ANIMATION

Grundsätzlich werden für eine Adventurespiel folgende Animationen benötigt:

- **Walkcircle**
- **Benutzen/Greifen Animation**
- **Sprachanimation**
- **Standanimation**

Je nach gewünschtem Detailgrad können natürlich noch zusätzliche, individuelle Animation für bestimmte Interaktionen erstellt werden. Zudem werden die Animationen des Charakters aus verschiedenen Perspektiven (von vorne, von hinten, seitlich) benötigt. Aus Zeitgründen wurden die Animationen für das vorliegende Projekt nur in einer Seitenansicht produziert.

Die Animationen für den Walkcircle und den Stand müssen aus einem *Loop* bestehen. Das heißt, das erste und das letzte Bild der Animation sind identisch, wodurch die Animation ohne merkbaren Übergang mehrmals hintereinander abgespielt werden kann. Andernfalls müsste beim Walkcircle zum Beispiel jeder Schritt einzeln animiert werden, was einerseits zeitlich nicht machbar wäre und zudem auch eine sehr hohe Anzahl an Einzelbildern zur Folge hätte, die im späteren Spiel zu lasten der Performance fallen würden.

Für die Produktion der Animation wurde der Hauptcharakter in 4 Teile zerlegt, sich überschneidende Teile wie Arme und Beine mussten dabei als separate Grafiken gespeichert werden, da die Marionetten-Animation hier nicht in der Lage ist, die Überschneidungen korrekt darzustellen.

Die einzelnen Grafiken wurden in After Effects als Bildmaterial geladen und eine neue Komposition erstellt. Um eine möglichst flüssige Bewegung zu erzeugen, wurde statt einer zeichentrick-üblichen Framerate von 12 Bildern pro Sekunde eine Framerate von 25 Bildern pro Sekunde verwendet. Diese entspricht in ihrer Bildfrequenz dem PAL Standard für Realfilm.

Um das korrekte Timing der Animation zu gewährleisten wurde zusätzlich eine Video einer gehenden Person als Referenz verwendet.

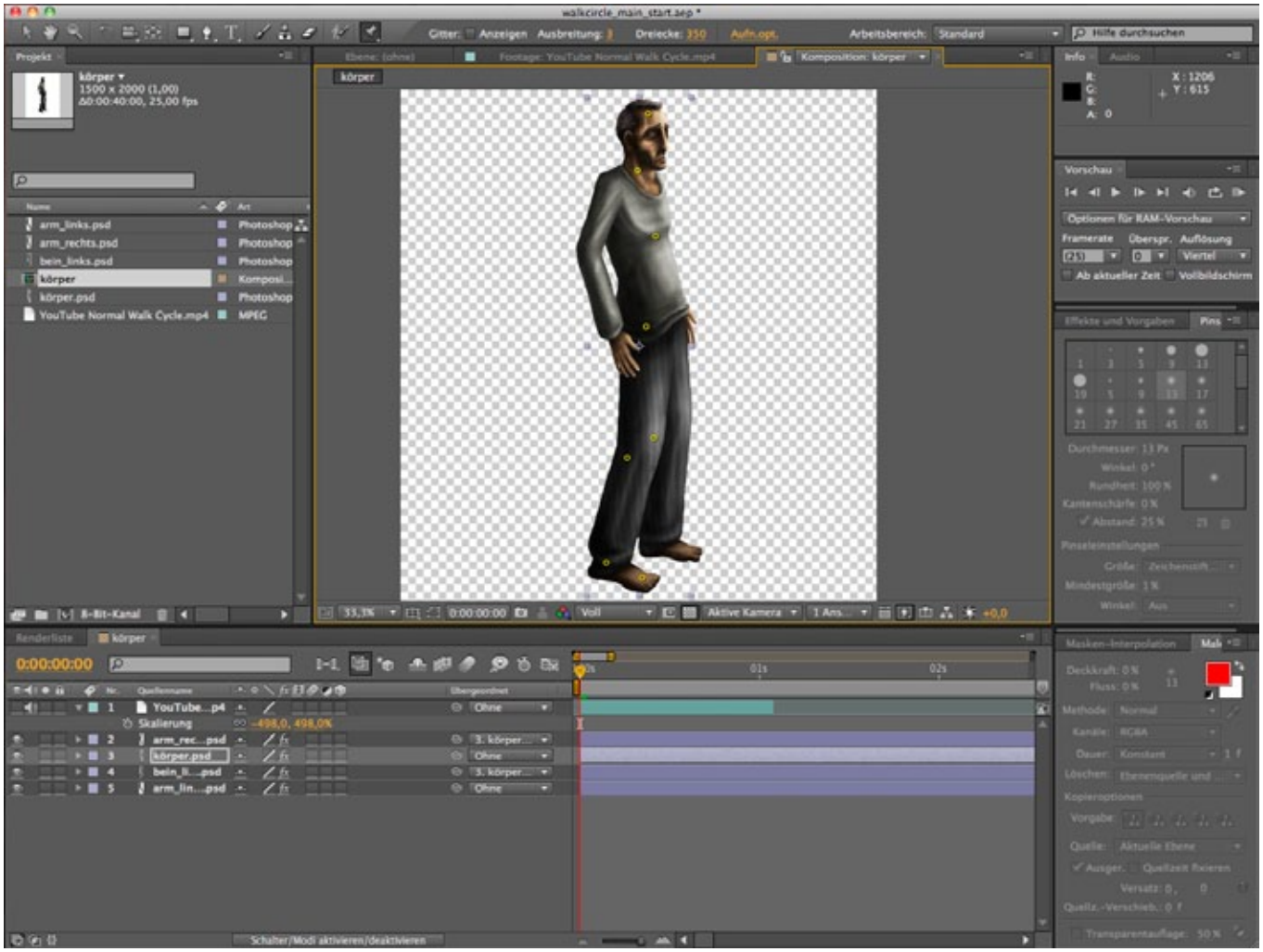
Innerhalb der Komposition wurden die einzelnen Elemente der Hauptfigur in einzelnen Ebenen übereinander gelegt. Als oberste Ebene wurde das entsprechende Referenzvideo für die Bewegung eingeladen und leicht transparent gestellt, so dass der darunterliegende Charakter noch erkennbar war.

Cutouts des Charakters

Fertig aufgebaute Komposition in After Effects.

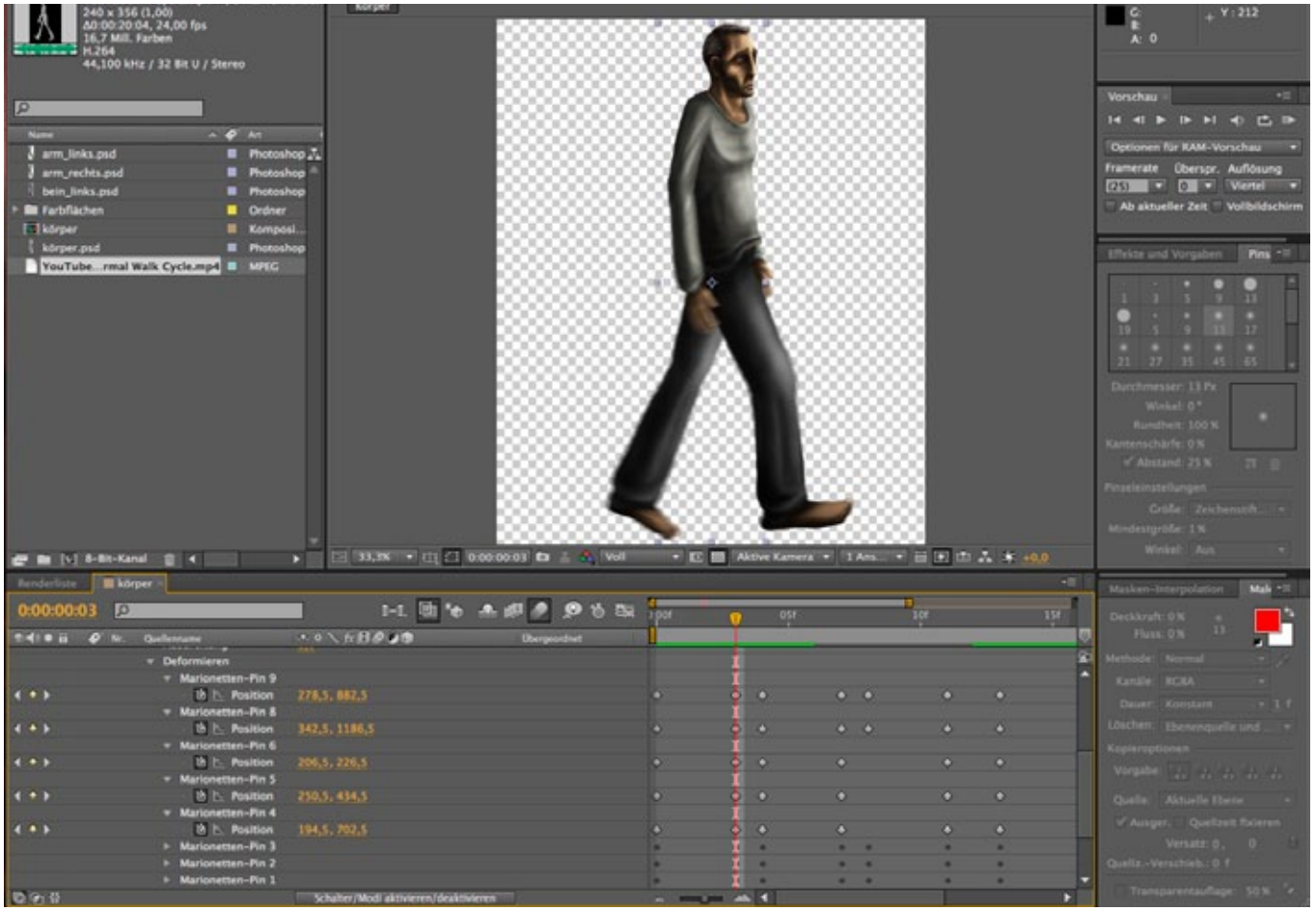
1. Referenzvideo
2. Charakter Arm rechts
3. Charakter Körper
4. Charakter Bein links
5. Charakter Arm links

Charakter mit eingeblendetem Referenzvideo



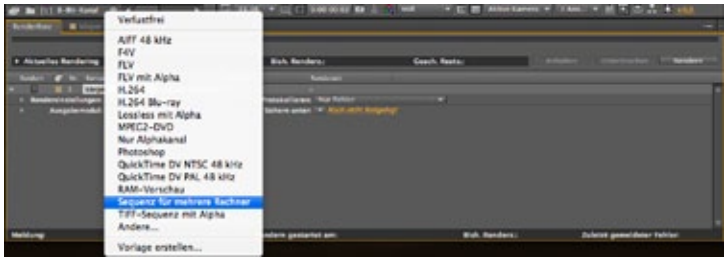
Nun wurden mit dem *„Marionetten-Werkzeug“* den Elementen des Charakters die benötigten Gelenkpunkte hinzugefügt. Dabei war es wichtig darauf zu achten, dass man möglichst wenig Gelenkpunkte verwendet. Die Animation des Charakters würde ansonsten erschwert und es könnte zu ungewünschten Verzerrungen der Grafiken kommen.

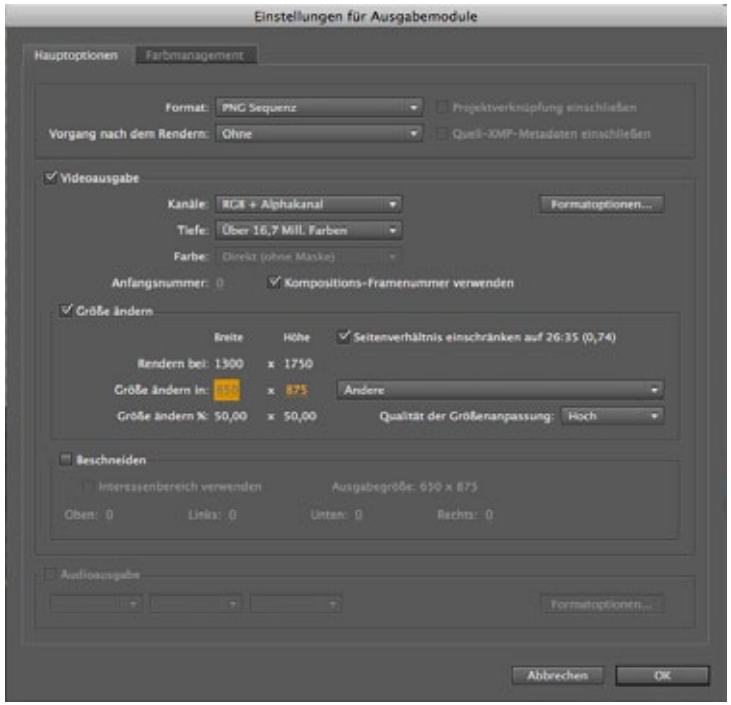
Nachdem die Gelenkpunkte angelegt waren, wurden zuerst die Extremposen der Bewegung, zum Beispiel ein angezogenes Bein des Charakters, festgelegt. Dazu wurde innerhalb der Timeline auf die Extrempose im Referenzvideo gespult und der Charakter über die Gelenkpunkte in die entsprechende Pose verzerrt. Die benötigten Keyframes werden dabei automatisch erzeugt. Durch diese Vorgehensweise wurde das korrekte Timing der Animation sichergestellt. Die Bewegungsstufen zwischen den einzelnen Extremposen werden dabei automatisch berechnet.



Um die Bewegung jedoch natürlich wirken zu lassen, ist eine zusätzliche Bearbeitung der sogenannten „Inbetweens“ notwendig. Dabei wurde auf entsprechende Zwischenbilder in der Animation gespult und die Pose des Charakters an das Referenzvideo angepasst.

Nachdem die Animation grob abgeschlossen war, ging es an das Feintuning der Bewegungen. Gemäß der am Anfang des Kapitels beschriebenen Animationsregeln ist eine Bewegung niemals linear, also mit gleichmäßiger Geschwindigkeit, sondern verläuft immer in einem Bogen. Das heißt die Bewegung startet langsam, beschleunigt dann und wird zum Übergang in die Extrempose wieder langsamer.
Dies wurde durch die Funktion *„Ease in“* und *„Ease Out“* an den entsprechenden Keyframes erreicht.





Alle 4 Animation (Gehen, Benutzen, Sprechen, Stand) wurden auf die gleiche Art und Weise animiert. Teilweise wurden jedoch die Gelenkpunkt je nach Animation anders gesetzt. So war für die Greif-Animation ein zusätzlicher Gelenkpunkt für die Finger des Charakters notwendig, da das Zugreifen der Hand ebenfalls animiert wurde, während dies beim Walkcircle nicht notwendig war.

Nachdem die Animationen fertiggestellt waren, wurden sie als *Bildsequenz* aus After Effects exportiert. Aufgrund des transparenten Hintergrunds der Figur wurde dabei das *PNG-Dateiformat* gewählt. Besonders zu beachten war dabei die Dateigröße der Einzelbilder, da diesem in fertigen Spiel teilweise in Echtzeit geladen werden müssen und es dadurch zu Performance Einbußen wie zum Beispiel *„Rucklern“* kommen kann. So bestand die Standanimation des atmenden Charakters zunächst aus 80 Einzelbildern mit einer Dateigröße von jeweils 200 KB was einer Gesamtgröße von 15 MB für diese Animation entsprach. Beim Laden der Animation brachte dies selbst leistungsstarke Computer zum Einbruch. Deshalb musste die Animation zu lasten der Qualität auf 50 Bilder beschränkt werden.

Fertige Bildsequenz des Walkcircles

ITEMS & INTERFACES

Das Gameplay eines Point & Click Adventures besteht aus dem Erkunden der Umgebung, dem Dialog mit anderen Spielfiguren, *„NPC“* genannt, die Kurzform für *Non-Playable-Characters*, sowie dem Lösen von Rätseln. Dafür müssen Gegenstände eingesammelt, benutzt und kombiniert werden. Zur Steuerung der Interaktionen werden Interfaces benötigt, durch die der Spieler die Handlungen der Spielfigur kontrollieren kann.

Seit der Entwicklung der Point & Click Adventures zu Beginn der 90er Jahre haben Sie die grundlegenden Aktionsverben nicht stark verändert. Standardmäßig werden die Aktionen *„Gehe zu“*, *„Betrachte“*, *„Nimm“*, *„Benutze“*, *„Spricht mit“* angeboten. In der Kombination ihrer Anwendung entstehen jedoch vielfältige Interaktionsmöglichkeiten.
Was sich jedoch weiterentwickelt hat, ist die Art wie diese Interaktionen durch den Spieler genutzt werden können. *„The Secret of Monkey Island, 1990, Lucasfilm Games“*, einer der Vorreiter der Point & Click Adventures, bot für die Interaktion eine lange Liste an Verben, aus denen der Spieler die gewünschte Interaktion auswählen musste. Eingesammelte Gegenstände wurden in einem *Inventar* durch die entsprechenden Bezeichnungen repräsentiert. So ergaben sich zum Teil aberwitzige Interaktionen wie zum Beispiel *„Benutze Gummihuhn mit Haken auf Stromleitung“*. Diese Art der Benutzerführung geht noch auf die klassischen Textadventures, dem quasi-Vorläufer des Point & Click Adventures zurück. Hier wurde komplett auf Grafiken verzichtet, der Spieler musste sich die Spielwelt alleine aus Beschreibung erschließen und Interaktionen über eine Liste von Verben steuern. Der Nachteil dieser Technik war, dass die Spieler teilweise sehr lange nach den richtigen Interaktionsmöglichkeiten suchen mussten.

Bereits der Nachfolger *„Monkey Island 2, 1991, Lucasfilm Games“* kam mit einer reduzierten Verbliste und Bildern als Repräsentation der Inventargegenstände aus. Die langwierige Suche nach den richtigen Interaktionsverben wurde somit verkürzt, jedoch noch nicht gänzlich beseitigt. Mit der Entwicklung von *„Sam & Max hit the Road, 1993, Lucasfilm Games“* und dem 3. Teil der Reihe *„The Curse of Monkey Island, LucasArts, 1997“* wurde das *kontextsensitive Interface* in die Point & Click Adventures eingeführt. Dabei ändert sich der Mauszeiger je nach Objekt in die entsprechend ausführbare Interaktion. Fährt man also mit dem Mauszeiger über ein aufnehmbares Objekt, wird automatisch der Befehl „Nimm“ eingeblendet.

The Secret Monkey Island
Interface und Inventar nehmen ca. 1/3 des gesamten Spielebildschirms ein. Interaktionsmöglichkeiten und Gegenstände werden durch Verben und Nomen dargestellt

Monkey Island 2
Die Dimension der Interfaces ist gleichbleibend. Die Verbliste ist jedoch verkürzt. Gegenstände im Inventar werden durch Bilder repräsentiert.

Monkey Island 3
Der gesamte Bildschirm wird nun zur Spieldarstellung. Der Mauszeiger zeit kontextsensitiv ausführbare Aktionen an. Das Inventar wird erst auf Knopfdruck eingeblendet.

Die Verbliste entfiel somit und der gesamte Bildschirm konnte für die Spielegrafik genutzt werden. Diese Art der kontext-sensitiven Steuerung hat sich bis heute im Adventure-Genre etabliert. Aktuelle Beispiele wie das Spiel *„Deponia, 2012, Deadalic“* verwenden genau diese Art der Steuerung. Die 2009 und 2011 veröffentlichten, grafisch überarbeiteten Special Editions von „The Secret of Monkey Island“ und „Monkey Island 2“ wurden vom Gameplay dieser Art der Interaktion angepasst.

INTERFACES

Gehe zu
Aktiv *Inaktiv*

Benutze/Nimm
Aktiv *Inaktiv*

Betrachte
Aktiv *Inaktiv*

Betrachte
Aktiv *Inaktiv*

ITEMS

Um den Spielern eine möglichst einfache Kontrolle über die Interaktionen zu bieten, wurde ebenfalls ein kontextsensitives Menu konzipiert. Die grundlegenden Interaktion bestehen aus *„Gehe zu“*, *„Benutze“*, *„Nimm“*, *„Betrachte“*, *„Sprich mit“*.

Um die Aktionen dem Spieler direkt zu visualisieren, wurden grafische Icons, repräsentiert durch die verwendeten Körperteile der Spielfigur Füße, Hände, Augen und Mund entworfen. Für ein direktes visuelles Feeback über die Interaktion wurden die Icons in zwei unterschiedlichen Zuständen angelegt, einem *aktiven* und einem *passiven*. Wählt der Spieler das Handsymbol aus, wird es in passiven Zustand durch eine geöffnete Hand dargestellt, fährt der Spieler damit über ein aufnehmbares Objekt schließt sich die Hand.

Damit die Icons optisch zum Spiel passen, wurden diese ebenfalls nach der in den vorangegangenen Kapiteln beschriebene Methode illustriert und in Photoshop koloriert. Beide Zustände wurden anschließend als PNG Grafiken mit transparentem Hintergrund exportiert.

Die aufnehembaren Gegenstände wurden ebenfalls auf diese Art und Weise erstellt.

SOUNDEFFEKTE UND MUSIK

Ein Spiel funktioniert, ähnlich wie ein film, nicht rein auf der visuellen Ebene. Ohne ein entsprechendes Sounddesign wirkt das Spiel leblos, erst die richtigen Töne vervollständigen die Illusion einer belebten Spielewelt. Ein entsprechender Soundtrack unterstreicht dabei die Atmosphäre der Szenerie und kann die Stimmung einer Szene maßgeblich beeinflussen.

Ein Bespiel für die Wirkung des richtigen Sounddesigns ist der Filmklassiker *„The Haunting, USA 1963, Robert Wise“*. Viele der furchteinflößenden Szenen werden hier alleine durch die entsprechende Soundkulisse aufgebaut. Auch die Filme von David Lynch funktionieren stark über das Zusammenspiel von Bild und Sound. Das Sounddesign schafft es hier, sogar in scheinbar harmlos wirkenden Szenen eine abgründige, bedrohliche Atmosphäre aufzubauen. Dies war letztendlich auch die Inspirationsquelle für die Musikauswahl im produzierten Spiel. Die Wahl fiel hierbei auf das Lied „Constant Fear“ der Jazzcombo „Bohren und der Club of Gore“.
Für die Soundeffekte wurde Geräusche aus freien Soundbibliotheken verwendet, zum Beispiel *„www.freesound.org“*.

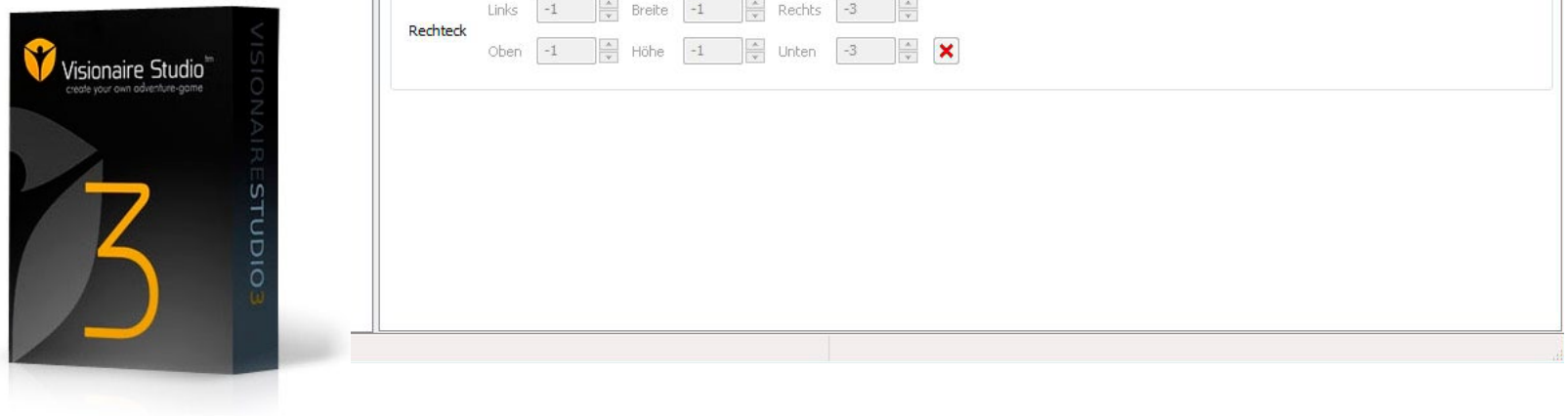
ARBEITEN MIT GAME-ENGINES

Als Game-Engines bezeichnet man Programme, welche die gesamten Inhalte das Spiels, also Grafiken, Sound, Animationen, Videos verwalten und durch entsprechende Programmierskripte die logische Struktur des Spiels aufbauen. Es ist sozusagen der „Motor“ der das ganze Spiel antreibt. Große Entwickler-Studios nutzen zur Spieleentwicklung oft eigens entwickelte Engines oder lizenzieren von anderen Entwickler-Studios entwickelte Engines. Die populärsten Beispiele für lizenzierbare Engines sind die *„ID Tech 4, id Software“*, *„Unreal Engine 3, Epic Games“* und die *„CryEngine 3, Crytek Studios“*. Die Kosten für eine solche Lizenz sind entsprechend hoch. Es gibt hier zwar auch die Möglichkeit eine kostenfreie Version zu erhalten, diese darf aber nicht für kommerzielle Projekte eingesetzt werden. Soll das auf dieser Basis entwickelte Spiel also anschließend verkauft werden, muss eine entsprechende Lizenz erworben werden.

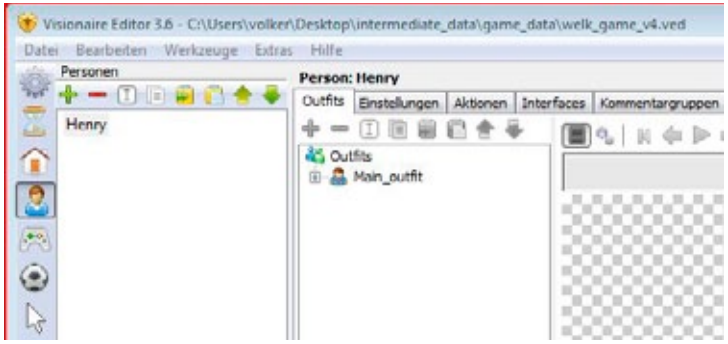
Da Indie-Entwicklern in der Regel weitaus weniger Budget für die Spieleentwicklung zur Verfügung steht, werden meist kostenlose bis sehr günstige Engines verwendet. Die bekannteste Engine ist in diesem Falle die *„Unity 3D Engine, Unity Technologies“*. Mit ihr lässt sich jedes Art von Spiel entwickeln, First-Person-Shooter, Rennspiele und Geschicklichkeitsspiele seien an dieser Stelle nur exemplarisch genannt.

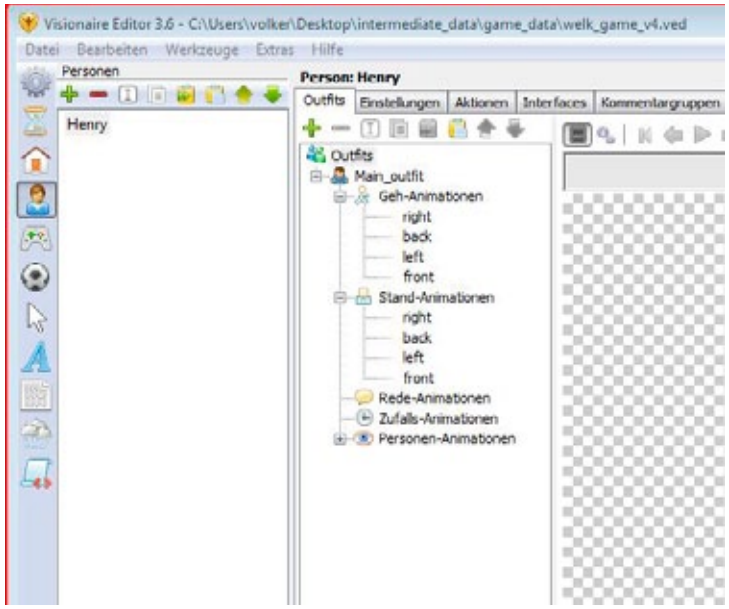
In diesem Projekt wird die speziell für Point & Click Adventures entwickelte Engine *„Visionaire Studio, siteface UG“* zur Erstellung des Spiels verwendet. Die Besonderheit dieser Engine ist, dass sie ausschließlich für die Entwicklung von Point & Click Adventures gedacht ist - andere Spielegattungen lassen sich damit nicht umsetzen. Dafür bietet sie jedoch ein großes Repertoire an vorgegebenen Aktionen, aus denen das Spiel zusammengesetzt werden kann. Im Gegensatz zu einer Engine wie Unity 3D sind keine Programmierkenntnisse notwendig, um funktionierende Interaktionen zu realisieren. Es ist jedoch möglich, über eine integrierte Scriptsprache zusätzlich eigene Funktionen zu entwickeln. Die nachfolgenden Seiten bieten einen Einblick in das Programm und zeigen Schritt für Schritt, wie der Prototyp des Spiels mithilfe der Engine umgesetzt wurde.

An dieser Stelle sei auf die sehr guten Tutorials zu Visionaire unter:
http://www2.visionaire2d.net/glenfx
verwiesen.
Diese dienten mir als Einstieg in die Software.


Visionaire Editor 3.6 - C:\Users\volker\Desktop\intermediate_data\game_data\welk_game_v4.ved
Datei Bearbeiten Werkzeuge Extras Hilfe
Spiel
Spieleinstellungen
Mauseinstellungen
Tastenaktionen
Sprachen
Spielinformationen
Spielübersicht
Erste Kulisse
Bedroom
Standardsprache
Unnamed
Auflösung
1280 x 1024
Zu Beginn folgende Aktion ausführen
Startaktion
Standardbefehl aktivieren
Nur nach erfolgreicher Ausführung
Firmenname
Spielname
Welk
Person bei sofortiger Ausführung ausrichten
Gegenstände aus dem Inventar können gezogen werden
Überspringen von Texten immer erlauben
Tasten-Aktionen während eines Dialogs ausführen
Interfaces in Menüs automatisch ausblenden
Personenbilder während Anzeige von Menüs im Speicher behalten
Aktive Person
Henry
Introfilm
Video auf Vollbild skalieren
Einblendverzögerung [ms]
600
Scrollgeschwindigkeit [Pixel/Sek]
300
Cachegröße für Grafiken [MB]
40
Textausrichtung
Zentriert
Ausrichtung der Sprechertexte
Links
Verkleinerungsfilter
Lineare Interpolation
Vergrößerungsfilter
Lineare Interpolation
Interaktion bei Personenanim. deaktivieren
Nicht deaktivieren
Schriftart für Objekttext
[leer]
Aktionstext
Anzeige
Aktionstext bei Cursor anzeigen
Schriftart
Font Henry neu
Rechteck
Links -1 Breite -1 Rechts -3
Oben -1 Höhe -1 Unten -3
Visionaire Studio
create your own adventure-game
VISIONAIRESTUDIO3

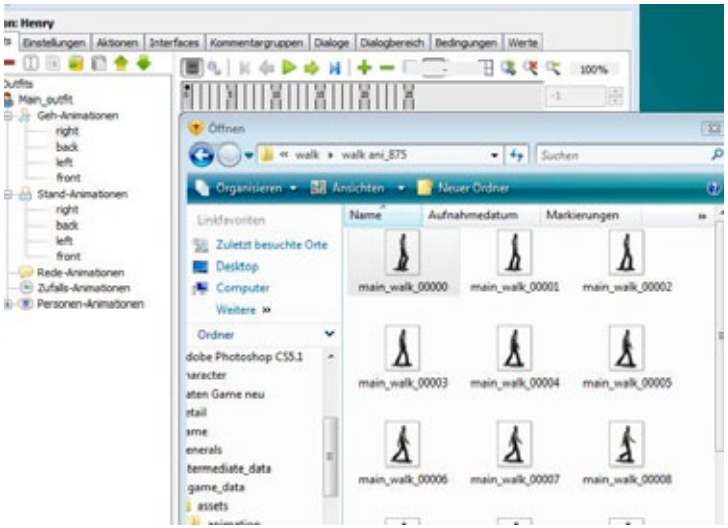
ANLEGEN VON CHARAKTEREN

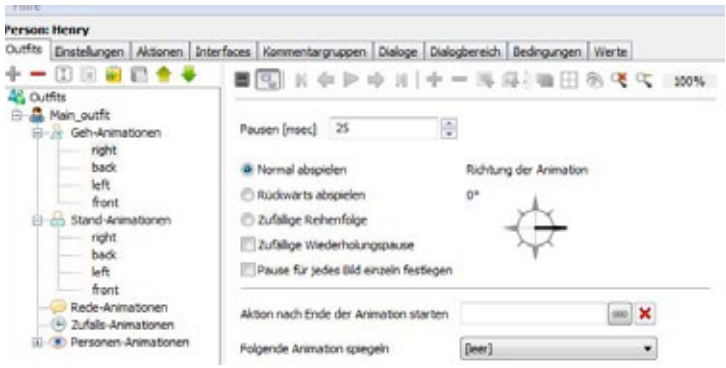


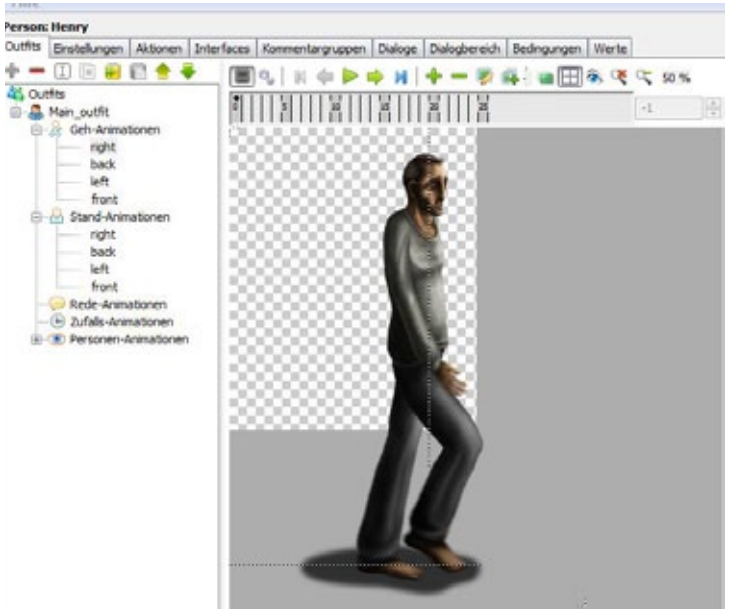


Unter dem Reiter *„Personen“* werden alle Charaktere des Spiels angelegt und verwaltet. Um den Hauptcharakter zu erstellen, muss über das *Symbol* ein neuer Charakter erstellt werden. Innerhalb des Personen-Fensters werden die Outfits des Charakters angezeigt. Outfits entsprechen dabei den verschiedenen Erscheinungsformen des Charakters. Es ist also möglich, mehrere Outfits für einen Charakter anzulegen, also zum Beispiel einen nackten Charakter und einen angezogenen. Die Outfits können später per Aktion im Spiel getauscht werden.

Innerhalb der Outfits existieren vorgefertigte Ordner für alle Animationen des Charakters. Als Standard sind dort die Ordner für die Standanimation, die Geh-Animation, Zufällige Animationen und spezielle Charakter-Animationen angelegt. Im Ordner „Zufällige Animationen“ können Animationen angelegt werden, die bei längerem Stillstand des Charakters abgespielt werden. Unter den Charakter-Animationen werden spezielle Animationen angelegt, die nur an bestimmten Punkten im Spiel verwendet werden, zum Beispiele eine Greif-Animation oder eine Sprech-Animation. Das Anlegen der Animationen ist immer nahezu identisch, deshalb wird hier nur das Anlegen der Geh-Animation beschrieben.







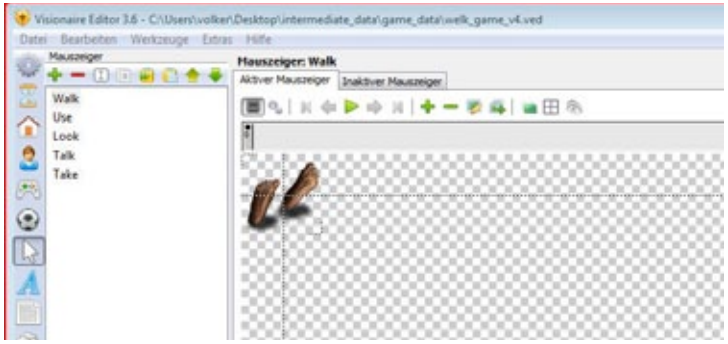
Innerhalb des Ordners Geh-Animationen existieren Unterordner für 4 unterschiedliche Ausrichtungen des Charakters, Nach Auswahl eines dieser Ordner können die Bilder des Charakters für die Animation eingeladen werden. Dabei kann sowohl ein *einzelnes Bild* als auch eine *Bildsequenz* eingeladen werden. Da sich die Figur bewegen soll, wird hier die Funktion Bildsequenz laden verwendet und die zuvor erstellte Bildsequenz des *Walk-circles* ausgewählt.

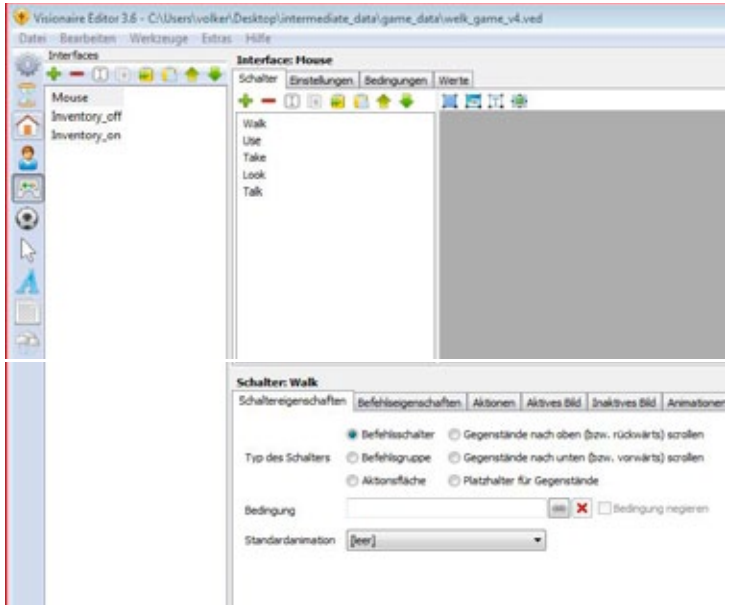
Die eingeladene Animation lässt sich im Vorschaufenster abspielen. Sollte die Animation zu schnell ablaufen, kann man dies unter *Einstellungen* korrigieren. Der im Feld *„Pausen“* angezeigte Wert entspricht der Frequenz, in dem zum nächsten Einzelbild der Animation gewechselt wird. Da für die Erstellung der Bildsequenz eine Framerate von 25 fps verwendet wurde, reicht hier ein Pausenwert von 25 aus. Sollte die Animation mit weniger Frames, zum Beispiel 12 Frames pro Sekunde erstellt worden sein, muss ein entsprechend höherer Wert eingetragen werden.

Anschließend muss noch der Fußpunkt des Charakter über das entsprechende *Icon* gesetzt werden. Dieser Schritt muss für alle Ausrichtungen des Charakters durchgeführt werden.

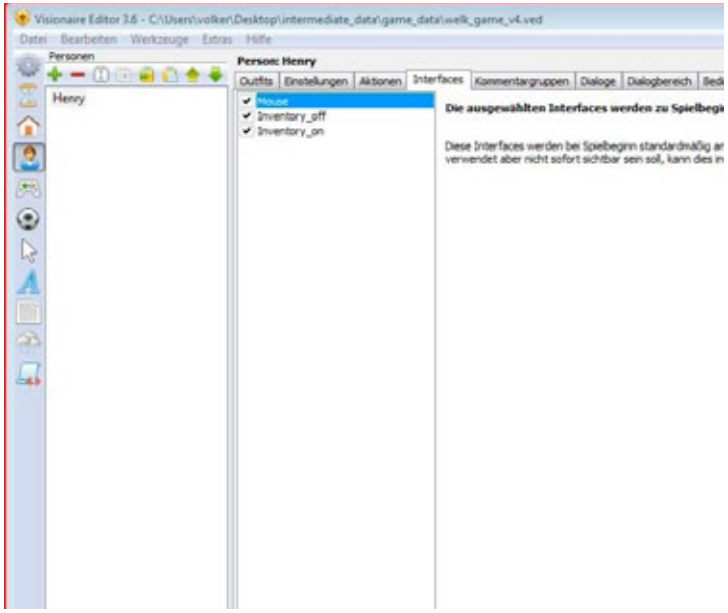
Nun können alle Animationen des auf diese Art angelegt werden. Manchmal reicht es aus, die Animation einer Richtung für eine andere nur zu spiegeln, zum Beispiel rechts und links.

MAUSZEIGER ERSTELLEN





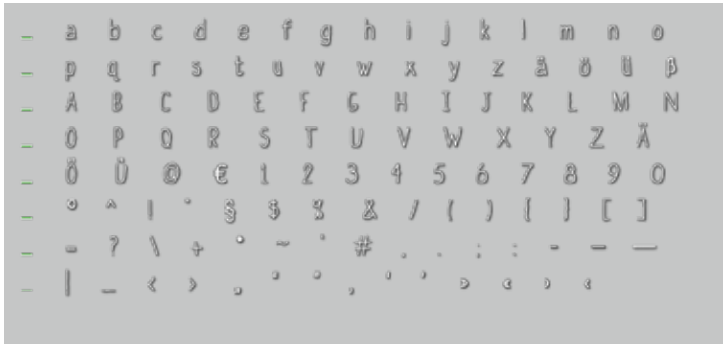
Um den Charakter zu steuern müssen entsprechende Bedienelemente angelegt werden. Unter dem Reiter *Mauszeiger* sollte jeweils ein Zeiger für *„Gehe zu“*, *„Benutze“*, *„Betrachte“* und *„Sprich mit“* angelegt werden. Anschließend können für jeden Zeiger über das Symbol die vorher erstellte Grafik für den aktiven und den inaktiven Zustand angelegt werden.
Es ist auch möglich, für den Mauszeiger eine Animation als Bildsequenz hinzuzufügen. Über das Icon sollte noch der Aktionsbereich des Zeigers angelegt werden. Dieser bestimmt, wo der Mauszeiger aktiviert wird.
Nun müssen die Mauszeiger noch in das Interface des Spiels eingebunden werden. Dazu wird unter dem Reiter *Interfaces* ein neues Interface mit dem Namen *„Mouse“* angelegt. Innerhalb des Interfaces muss für jede Aktion ein Schalter angelegt werden. Innerhalb der *Schaltereigenschaften* muss jeder der Schalter als *Befehlsschalter* definiert werden. In den Befehlseigenschaften wird festgelegt, wie sich der Schalter verhält. Hier sollte der Name des Schalter vergeben werden (Benennung beliebig), der entsprechende Cursor ausgewählt werden, sowie festgelegt werden, womit der Schalter benutzt werden kann. In der Regel reicht hier die Einstellung Personen und Objekte und die Einstellung *„Normaler Befehl“*. Nur beim „Benutzen“-Schalter ist es empfehlenswert die Option *„Gegenstände können mit diesem Befehl gezogen werden“* zu aktivieren.

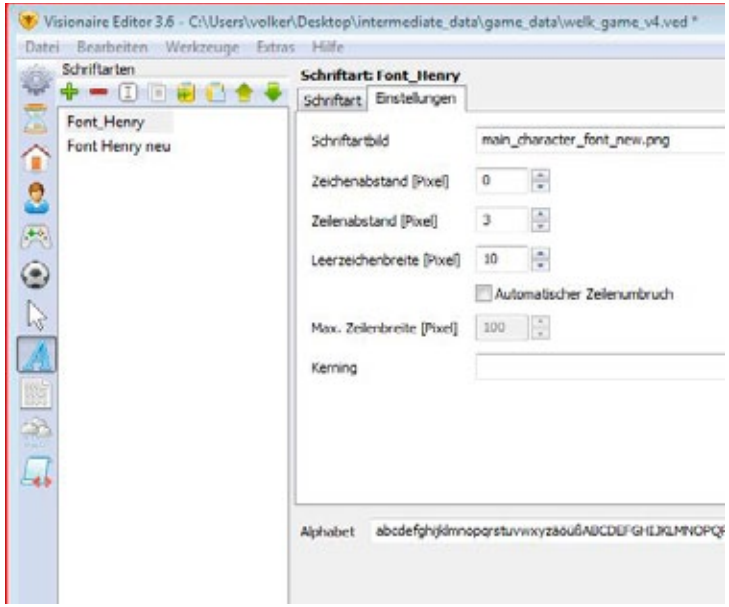


Zudem sollte dieser Schalter als *„Kombinierter Befehl“* definiert werden, damit Gegenstände miteinander benutzt werden können.

Nachdem das Interface erstellt wurde, muss es noch dem Spielcharakter zugeordnet werden. Dafür muss unter dem Reiter *„Personen“* der Hauptcharakter ausgewählt werden und unter der *„Interfaces“* das Interface „Mouse“ mit einem Haken versehen werden.

SCHRIFTEN



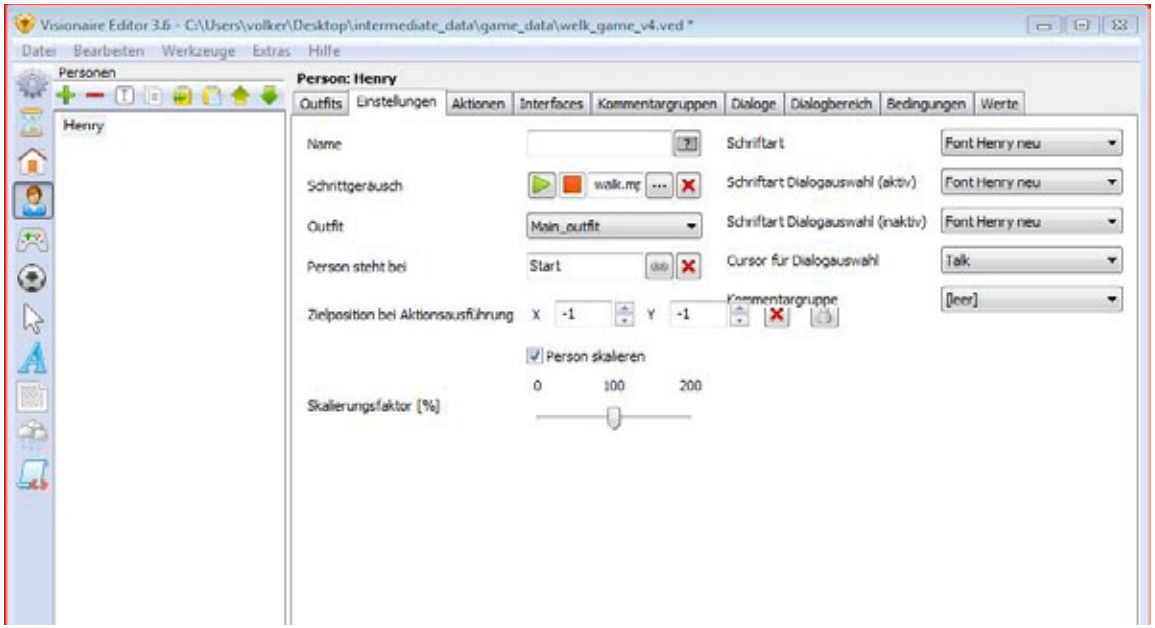


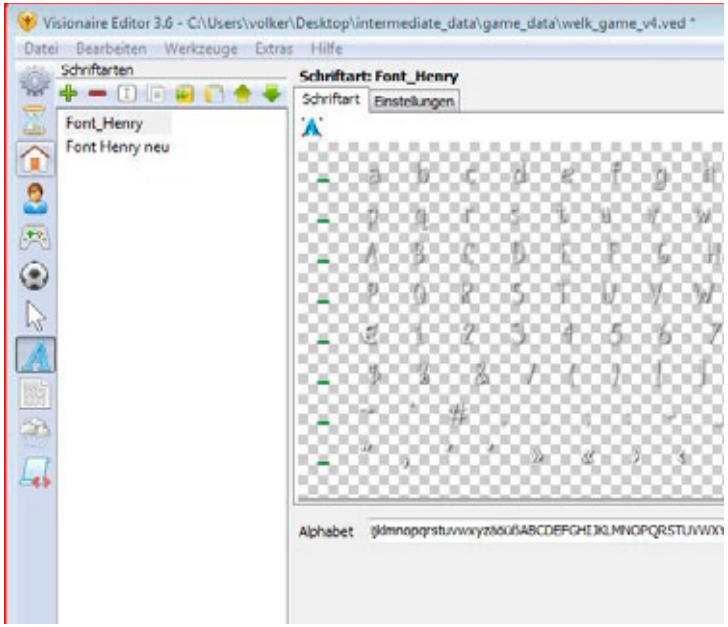
Für alle Texte im Spiel werden Schriftbilder benötigt. Dabei macht es Sinn, unterschiedlichen Charakteren eigene Schriftbilder für Ihre Dialogzeilen zu geben, damit der Spieler die einzelnen Dialogzeilen den Charakteren zuordnen kann.
Da in diesem Projekt nur ein Charakter verwendet wird, reicht ein Schriftbild aus.

Dafür wurde zunächst unter dem Reiter *„Schriften“* eine neue Schriftart angelegt und benannt. Innerhalb der Schriftart wurde unter *„Einstellungen“* das entsprechende *Schriftbild* als PNG mit Alphakanal geladen. Bei der Erstellung des Schriftbildes ist es wichtig, dass die einzelnen Buchstaben klar voneinander getrennt sind und auf einer Grundlinie liegen.

Die grünen Linien dienen dem Programm zur Orientierung und sind wichtig, damit die einzelnen Buchstaben richtig erkannt werden.

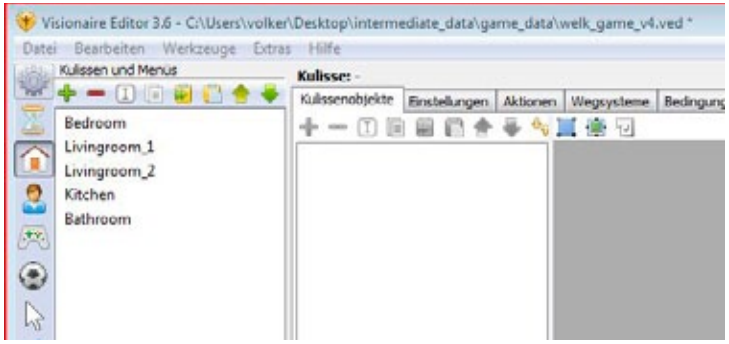
In dem Feld *„Alphabet“* müssen die Buchstaben in der gleichen Reihenfolge von rechts nach links angegeben werden, wie sie im Schriftbild vorkommen. Anschließend sollte das Programm automatisch alle Buchstaben richtig zuordnen. Soll der Abstand zwischen den einzelnen Buchstaben im Spiel größer oder kleiner sein, lässt sich dies in den entsprechenden Feldern einstellen.

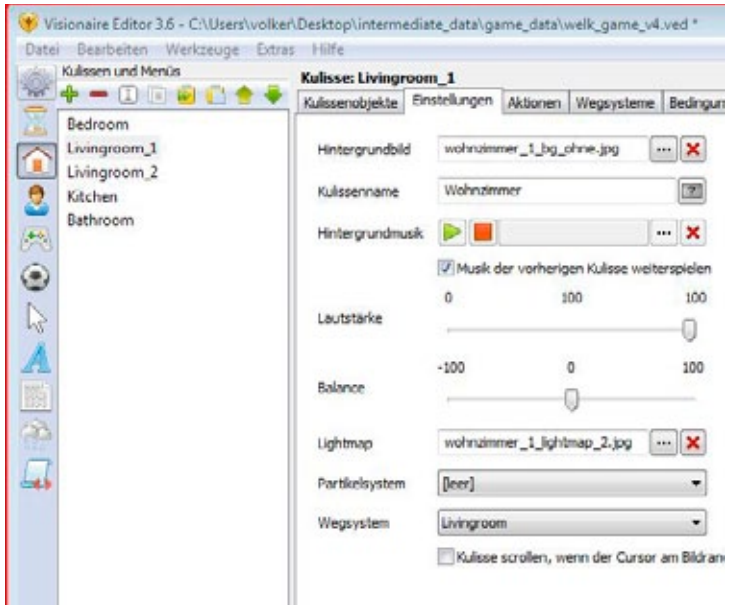




Die fertige Schrift muss noch dem Charakter zugeordnet werden. Dazu wurde im Reiter „*Personen*“, der entsprechende Charakter ausgewählt und ihm im Feld „*Einstellungen*“ die gewünschte Schriftart zugewiesen.

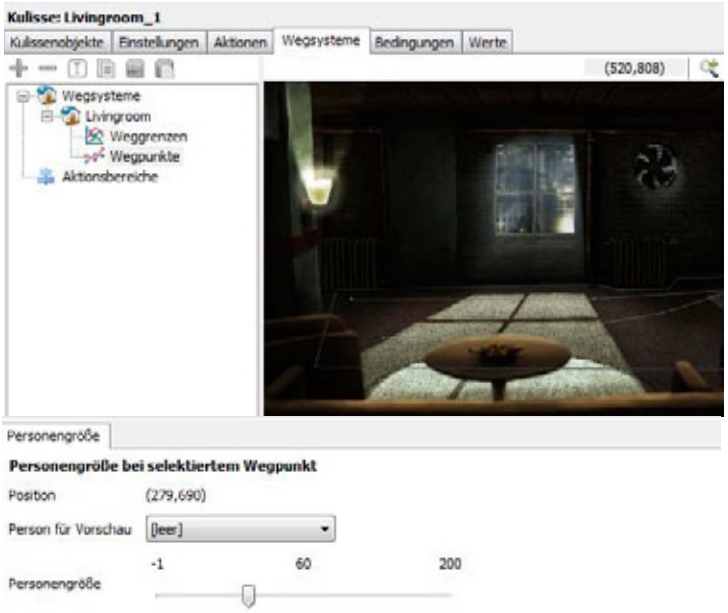
KULISSEN





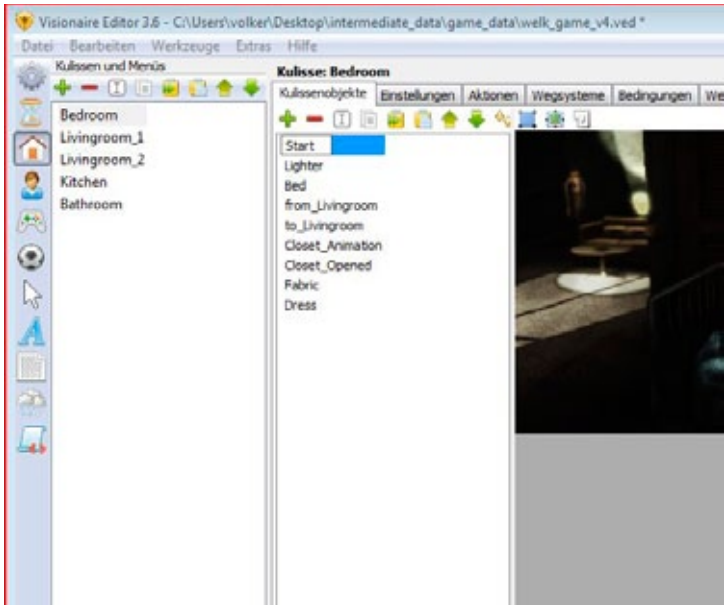
Alle Umgebungen im Spiel, also die Räume in denen sich der Charakter bewegt, werden als Kulissen bezeichnet.

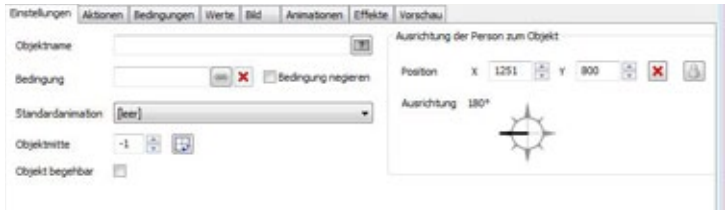
Über den *+ Button* kann man beliebig viele Kulissen erstellen. In diesem Projekt werden 5 Kulissen benötigt. In jeder Kulisse sollte zunächst unter „Einstellungen“ das vorbereitete Hintergrundbild geladen werden. In diesem Fenster lässt sich ebenfalls eine Hintergrundmusik für die Kulisse sowie die entsprechende Lightmap, also die Grafik für die Beleuchtung der Szene, auswählen.



Damit sich der Charakter innerhalb der Kulisse bewegen kann, werden *Weggrenzen* und *Wegpunkte* benötigt. Beide werden unter dem Punkt *„Wegsysteme“* zusammengefasst. Bei einer neu angelegten Kulissen muss zunächst über den *+ Button* ein neues Wegsystem angelegt werden. Weggrenzen umrahmen den Bereich, in dem sich der Charakter in der Kulisse bewegen kann. Wegpunkte geben den Pfad an, auf dem sich der Charakter durch die Kulisse bewegt. Beide müssen manuell gesetzt werden.

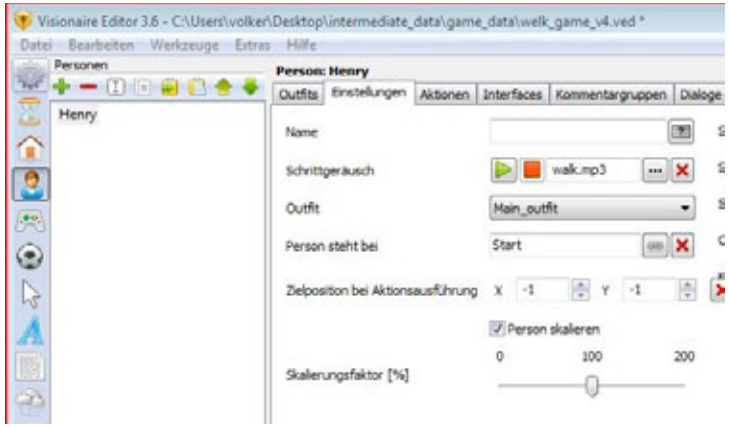
Damit sich die Größe des Charakters je nach Position im Raum ändert, können die Werte für die Skalierung an den entsprechenden Wegpunkten eingetragen werden.



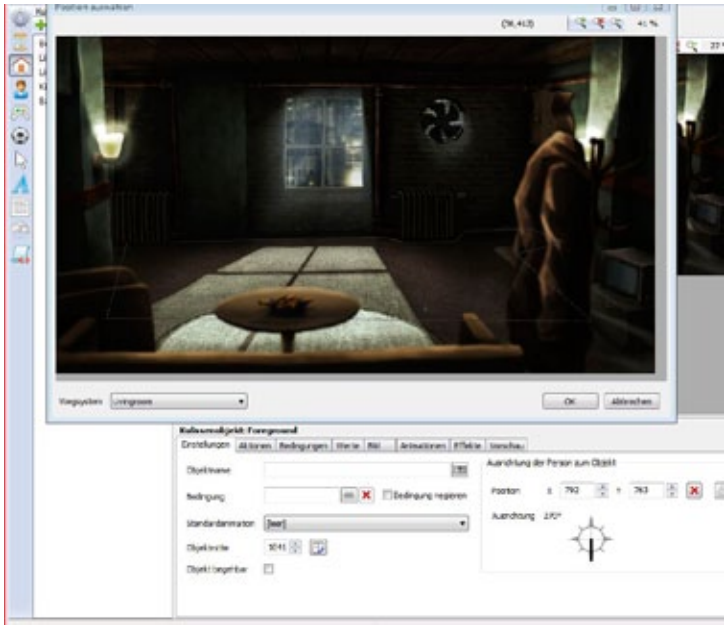


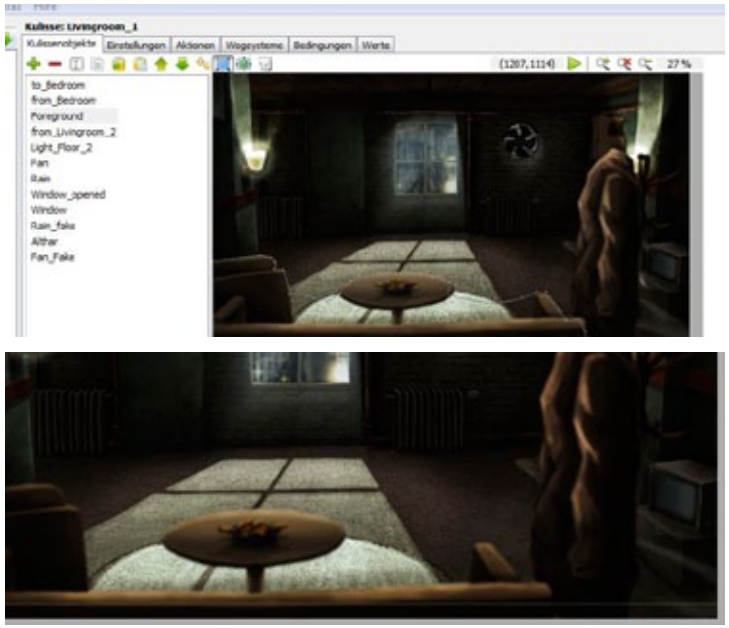
Startposition des Charakters festlegen

Um die Startposition des Charakters festzulegen, muss in der ersten Kulisse des Spiels ein neues Kulissenobjekt erzeugt werden. Die geschieht innerhalb der Kulisse unter dem Reiter *„Kulissenobjekte“* durch ein Klick auf den *+ Button*. Das neu entstandene Objekt kann *„Start“* genannt werden. Über den Button *„Objektposition setzen“* wird die Position des Kulissenobjekts bestimmt. Damit wird jedoch nicht die tatsächliche Position des Objekts beschrieben, sondern die, zu der sich die Spielfigur bewegt, wenn sie mit dem Objekt interagiert. Dabei ist darauf zu achten, dass sich die Position innerhalb der festgelegten Weggrenzen befindet, da der Charakter diese ansonsten nie erreichen kann. Über den darunterliegenden *Kompass* lässt sich die Ausrichtung der Spielfigur an der Festgelegten Objektposition bestimmen.



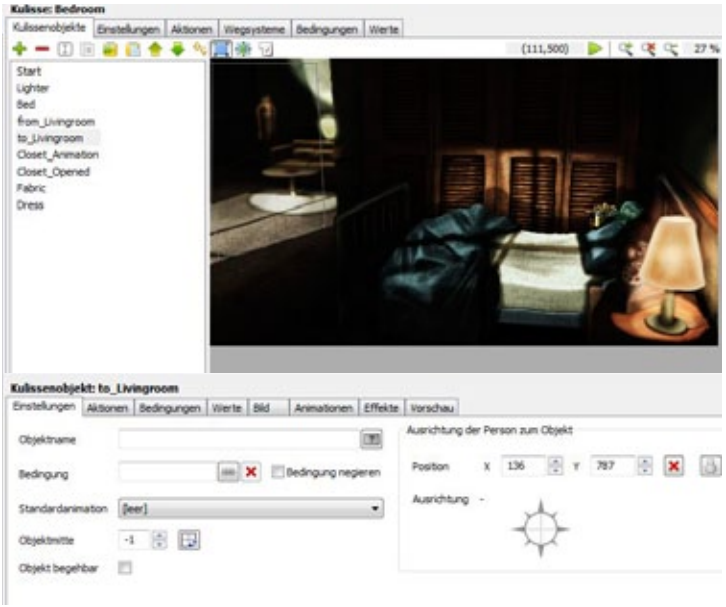
Um den Charakter die Startposition zuzuweisen, wechselt man wieder auf den Reiter *„Personen“* und wählt unter „Einstellungen“ die Kulisse und den Startpunkt im Feld *„Person steht bei“* entsprechend aus.





Weitere Kulissenobjekte hinzufügen

Alle Objekte mit denen der Charakter interagieren kann, müssen als Kulissenobjekte angelegt werden. Innerhalb eines Kulissenobjekts können über den Reiter *„Bild“* oder *„Animation“* die Bilder für das Erscheinungsbild des Objekts eingeladen werden. Diese können in der Kulisse über die Funktion *„Bilder und Animationen platzieren“* innerhalb der Szene positioniert werden. Anschließend muss wiederum die Objektposition gesetzt werden und die Ausrichtung des Charakter festgelegt werden. Über die Funktion *„Objektmitte festlegen“* wird dem Objekt eine Linie zugeordnet, die angibt, wann sich der Charakter vor oder hinter einem Objekt befindet. In dem vorliegenden Beispiel der Couch im Vordergrund wurde die Linie so gesetzt, dass sich der Charakter immer hinter dem Objekt befindet. Damit die Spielfigur das Objekt anwählen kann, muss noch der *Objektbereich* definiert werden. Die Kontur beschreibt den Bereich, in dem der Spieler das Objekt mit dem Mauszeiger anwählen kann. Wird er nicht gesetzt, kann der Spieler nicht mit dem Objekt interagieren. Im Gegensatz dazu kann aber ein Objektbereich definiert werden, ohne dass dem Kulissenobjekt ein eigenes Bild zugewiesen werden muss. So lassen sich Elemente aus dem Hintergrundbild der Kulisse umrahmen und mit einer Interaktion versehen. Diese Technik kommt im nächsten Schritt zum Einsatz.

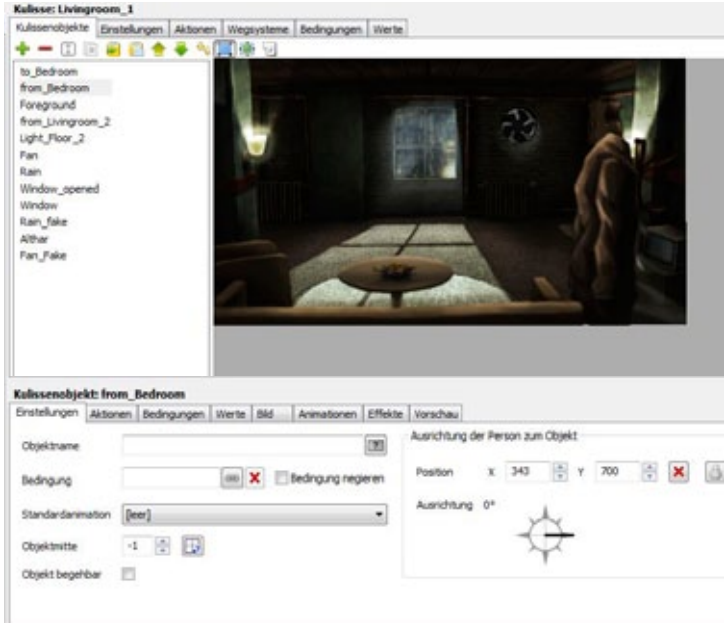


Übergänge zwischen den Kulissen erstellen.

Damit die Spielfigur die Kulissen wechseln kann, müssen Übergänge geschaffen werden. Dazu werden in zwei benachbarten Kulissen jeweils ein neues *Kulissenobjekt* wie zuvor beschrieben erstellt.

Anschließen wird dem Kulissenobjekt der Ausgangs-Kulisse in dem Reiter *„Aktionen“* eine entsprechende Aktion zugewiesen. Über das *+ Symbol* wird diese neue Aktion angelegt. Die Befehlsart lautet dabei *„Befehl auf Objekt angewendet“*. Als Befehl wird in dem vorliegenden Beispiel *„Walk“ (Gehe zu)* verwendet, da der Kulissenwechsel einfach über den Boden am Rand der Kulisse gesteuert werden soll. Wird der Kulissenwechsel zum Beispiel über eine Türe aufgerufen, macht der Befehl *„Benutze“* mehr Sinn.

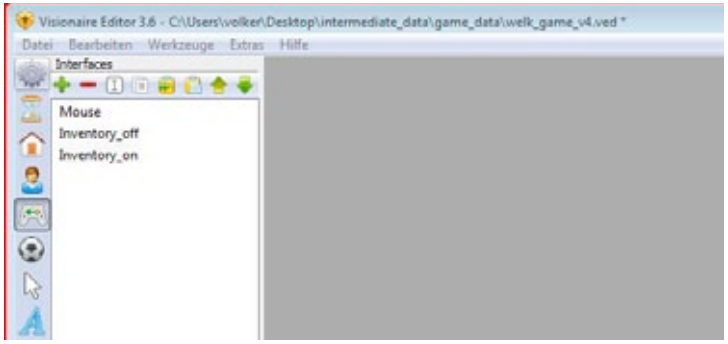
Anschließend muss über das *+ Symbol* im Aktionsfenster die Aktion *„Kulisse wechseln“* hinzugefügt und ausgewählt werden. Im Feld *„Person steht bei“* wird das entsprechende Kulissenobjekt in der nächsten Kulisse ausgewählt. Nun kann noch festgelegt werden, ob die nächste Kulisse sofort erscheinen oder langsam eingeblendet werden soll. Da im vorliegenden Projekt der Kulissenwechsel mehr die Anmutung einer neuen Kameraeinstellung haben sollte, wurde eine sofortige Einblendung der neuen Kulisse verwendet.

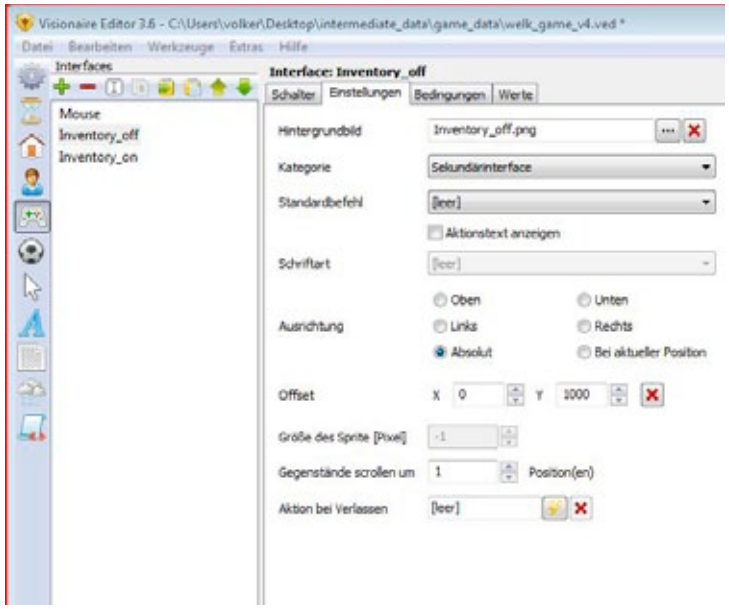


Damit die Spielfigur auch wieder in die erste Kulisse zurückkehren kann, muss die gesamte Aktion im Umkehrschluss auch für das Kulissenobjekt im zweiten Raum angelegt werden. Pro Raum lassen sich beliebig viele solcher Kulissenwechsel erstellen.

Sind nun alle Kulissen auf diese Art miteinander verbunden, kann der Spieler bereits den gesamten ersten Level erkunden. Um Interaktionen zu erzeugen werden aber auch aufnehmbare Objekte und somit auch ein Inventar benötigt.

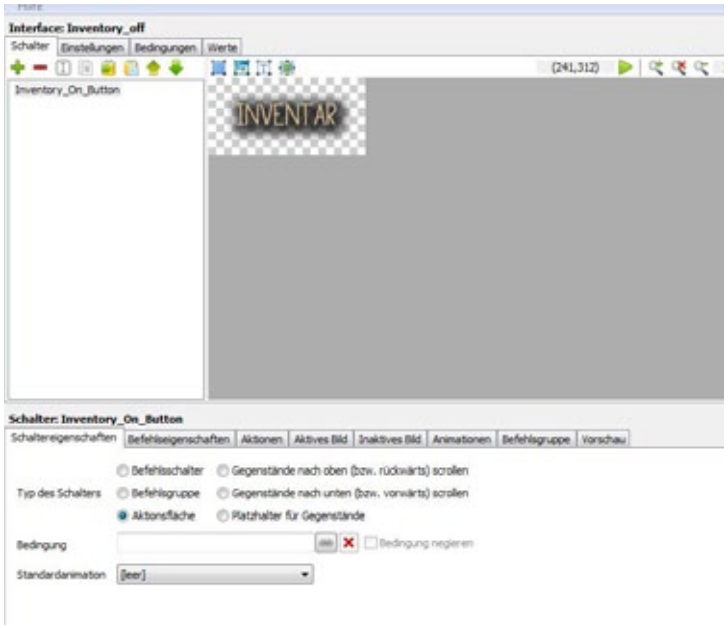
ANLEGEN DES INVENTARS

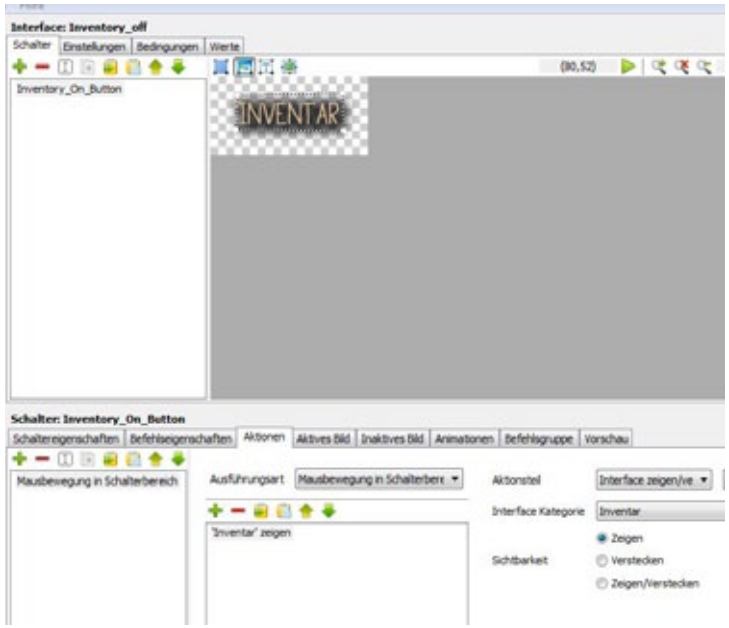


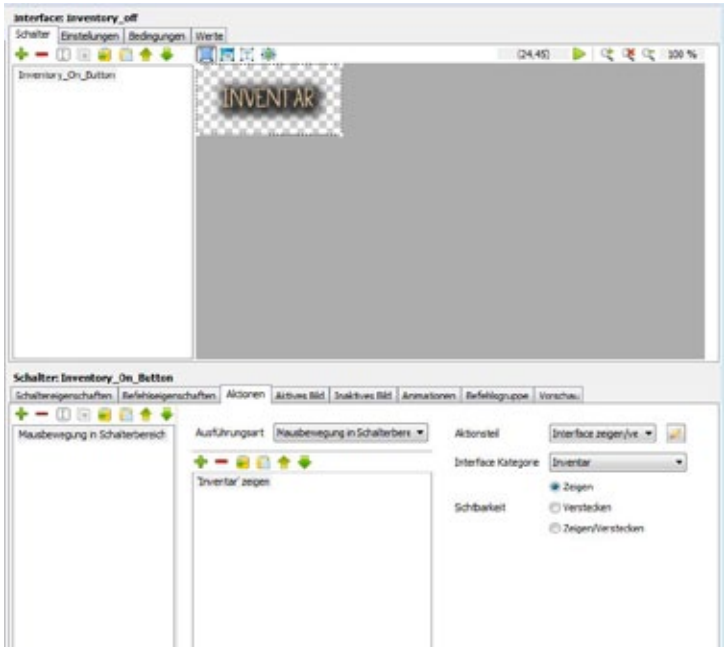


Damit Gegenstände im Spiel eingesammelt werden können, wird ein Inventar benötigt. In diesem werden alle aufgenommenen Gegenstände abgelegt.

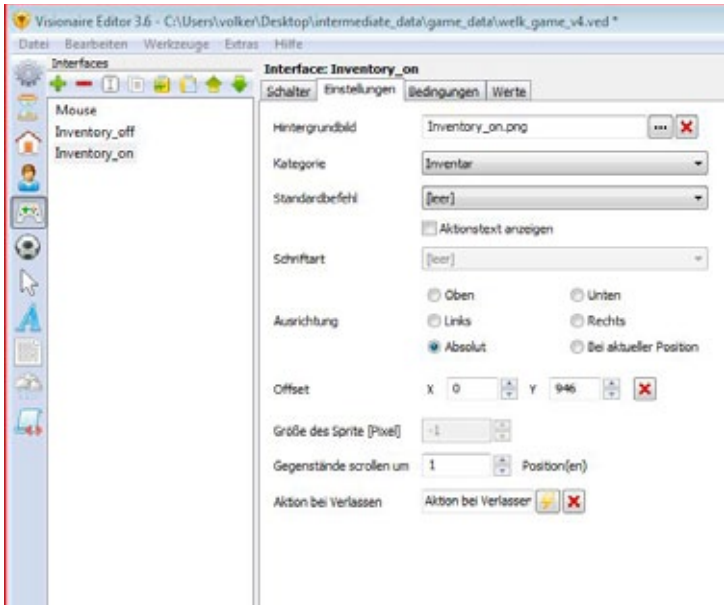
Dafür müssen im Reiter *„Interfaces“* zwei neue Interfaces angelegt werden. Eines, exemplarisch *„Inventory off“* genannt, dient als Button um das eigentliche Interface aufzurufen. Das *„Inventory on“* genannte Interface stellt das Inventar dar.
Im Button „Inventory off“ kann im Reiter Einstellungen ein entsprechendes Bild für den Button eingeladen werden. Die Klasse sollte hier als *„Sekundär-Interface“* angelegt werden.
Im Reiter *„Schalter“* muss nun ein neuer Schalter angelegt werden. Ähnlich der Kulissenobjekte muss auch die der Objektbereich gesetzt werden. Zusätzlich muss auch ein Interface-Bereich angelegt werden.
Unter Schaltereinstellungen muss der Schalter-Typ „Aktionsbereich“ gewählt werden. Anschließend wird dem Schalter eine Aktion zugewiesen. Da sich das Inventar öffnen soll, sobald der Mauszeiger über die Schaltfläche fährt, wird der Aktions-

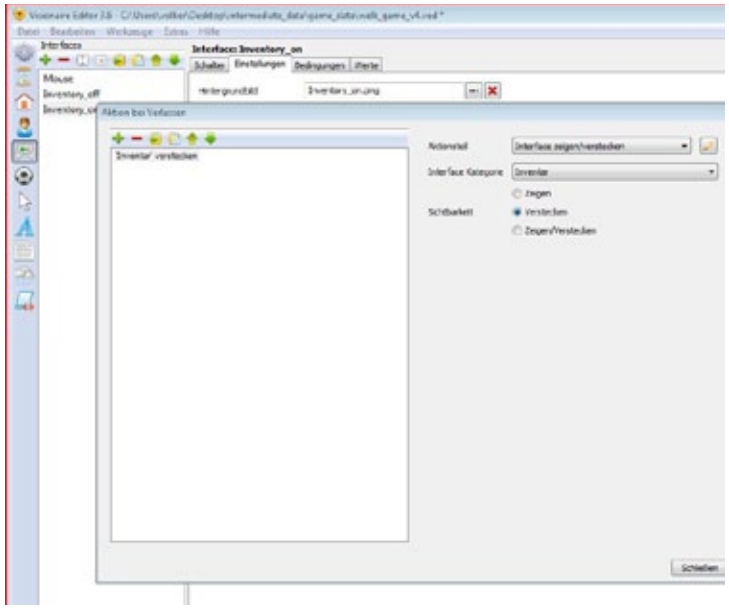






Typ *„Mausbewegung in Objektbereich“* gewählt. Als Aktion wird Interface zeigen/verstecken aus der Liste gewählt und als Interface Klasse „Inventar“ eingestellt. Die Sichtbarkeit wird auf „Anzeigen“ gestellt.



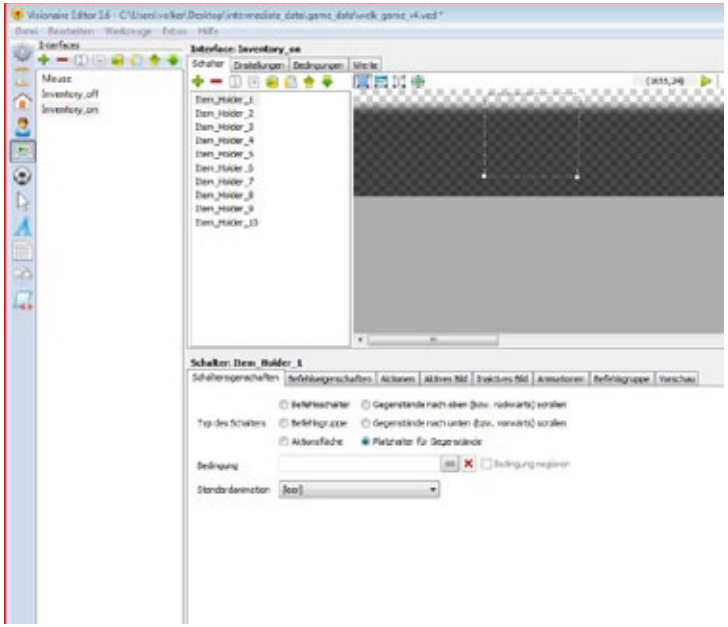


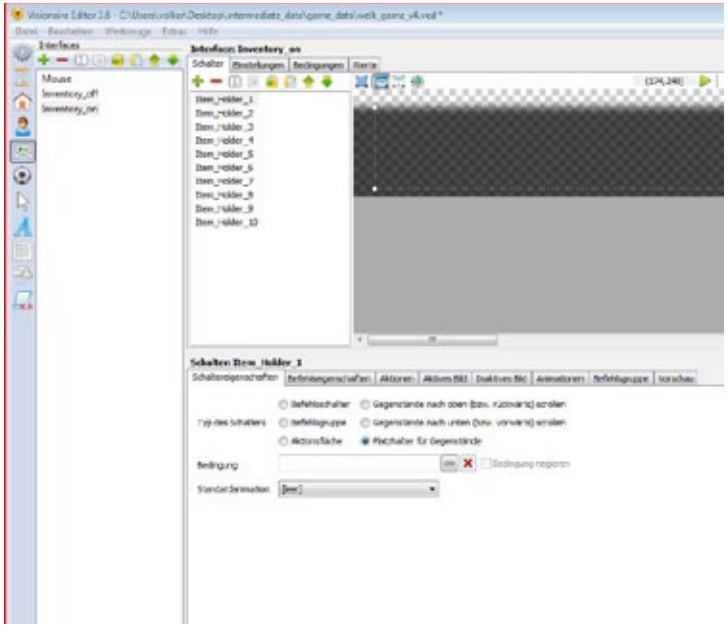
Nun kann das eigentliche Inventar, also *„Inventory on“* bearbeitet werden. In den Einstellungen wird wie bekannt das Hintergrundbild für das Inventar geladen. Als Klasse muss diesmal *„Inventar“* gesetzt werden. Damit das Inventar verschwindet, sobald man es mit dem Mauszeiger verlässt, muss hier noch eine entsprechende Aktion hinzugefügt werden und zwar *„Aktion beim verlassen“* ⇢ *„Interface zeigen/verstecken“* ⇢ *„Verstecken“*. Aktion bei Verlassen Aktion bei Verlasser

Unter dem Reiter *„Schalter“* werden nun die Platzhalter für die Objekte im Inventar angelegt. Dazu werden für jeden Objektplatz ein neuer Schalter vom Typ *„Platzhalter für Objekte“* angelegt. Über den Objektbereich wird wiederum die Position und die Größe des Platzhalters festgelegt.

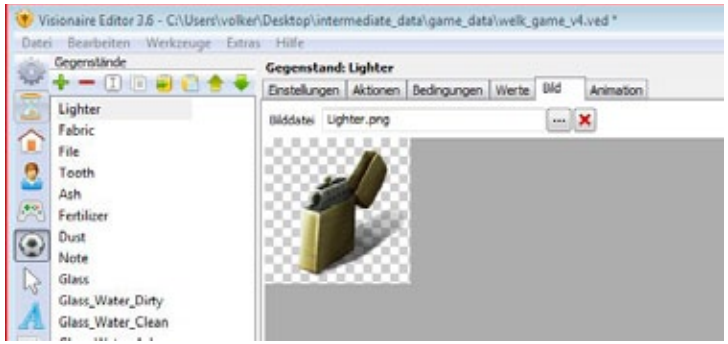
Je nach Umfang des Spiels werden unterschiedlich viele Platzhalter benötigt. In diesem Projekt reichten zehn aus. Hier sollte man sich auf jeden Fall im Klaren sein, wie viele Objekte die Spielfigur maximal bei sich tragen kann. Abschließend muss noch der Interface-Bereich festgelegt werden. Dieser muss alle Objekt-Platzhalter umschließen.

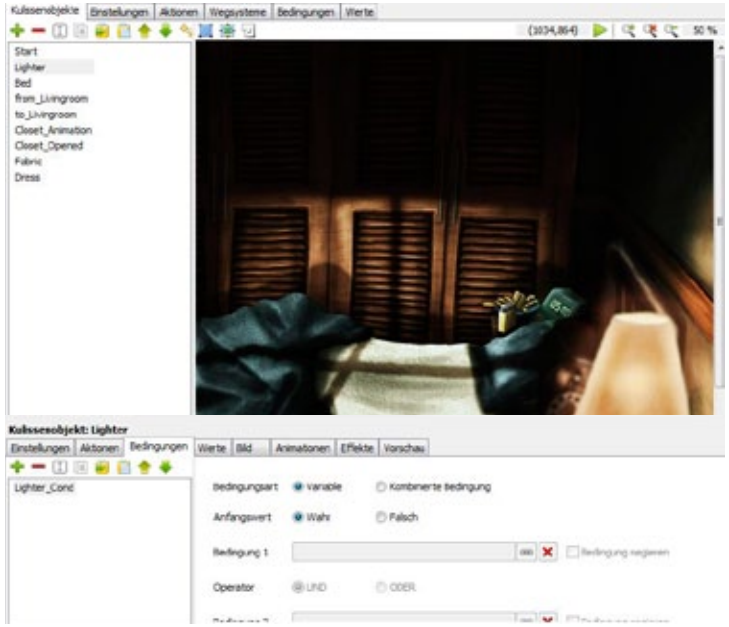
Entsprechend der Maus-Interfaces muss auch das Inventar der Spielfigur im Reiter *„Personen“* zugeordnet werden.





ITEMS



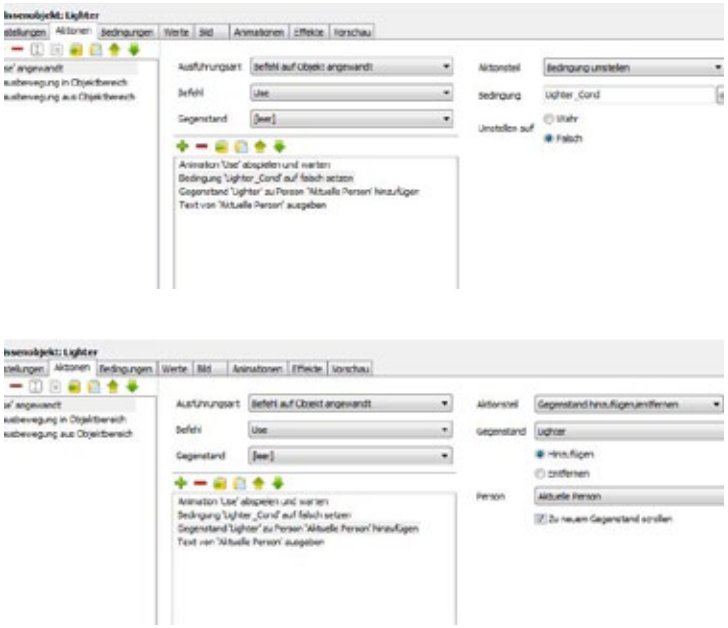


Zuletzt werden die Gegenstände des Spiels angelegt. Dazu werden im Reiter *„Gegenstände“* entsprechend viele neue Gegenstände angelegt. Diese entsprechen den Repräsentationen der Gegenstände im Inventar. Für jeden Gegenstand sollte im Reiter *„Bild“* nun das entsprechende Objektbild geladen werden.

Damit der Gegenstand mitgenommen werden kann, muss er noch in der entsprechenden Kulisse platziert werden. Dazu wählt man im Reiter *„Kulissen“* die entsprechende Kulisse aus und erstellt eine neues Kulissenobjekt - an dieser Stelle exemplarisch das Feuerzeug. *Objektbild, Objektbereich* und *Position* müssen wie gewohnt definiert werden.

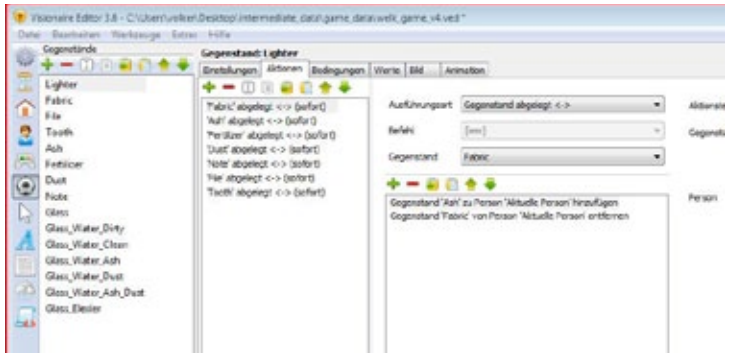
Neu ist die Verwendung einer *Bedingung*. Bedingungen steuern die Sichtbarkeit eines Objekts. Steht die Bedingung auf *„Wahr“* ist das Objekt sichtbar, bei *„Falsch“* ist es unsichtbar. Dies dient dazu, Objekt verschwinden zu lassen, nachdem der Spieler sie aufgenommen hat. Bei dem Beispiel-Objekt *„Lighter“* wird also die Bedingung *„Lighter_Visible“* erstellt. Die Benennung hat dabei keine Auswirkungen auf die Bedingung. Da das Feuerzeug zu Beginn sichtbar sein soll, wird die Bedingung auf *„Wahr“* gestellt. Im Reiter *„Einstellungen“* des Kulissenobjekts muss die Bedingung noch in dem entsprechenden Feld zugeordnet werden.

Nun kann die entsprechende Aktion für das Aufnehmen des Objekts erstellt werden. Dazu wird im Kulissen-Objekt Lighter die Aktion Befehl auf Gegenstand angewendet hinzugefügt. Als Befehl wird „Use“ (Benutze) verwendet.



Im Aktionsfenster werden folgende Aktionen angelegt:
Gegenstand hinzufügen ⇢ Lighter
(Dadurch wird der Gegenstand dem Inventar hinzugefügt)
Bedingung umstellen ⇢ Lighter_visible ⇢ Falsch
(Dadurch verschwindet der Gegenstand aus der Kulisse)
Text von ausgeben ⇢ „Ich habe das Feuerzeug aufgenommen“
(Dieser Text wird von der Spielfigur gesprochen und dient den Spielern als Feedback das der Gegenstand erfolgreich aufgenommen wurde.

Falls in den Personen-Animation eine Spezial-Animation für das Aufnehmen von Gegenständen erstellt wurde, kann diese auch an dieser Stelle aktiviert und abgespielt werden. Dazu muss zusätzlich noch die Aktion *„Animation abspielen“* hier hinzugefügt werden.



Gegenstände kombinieren und benutzen

Um Gegenstände miteinander zu kombinieren müssen in dem Reiter „Gegenstände“ die entsprechenden Aktionen angelegt werden. Für die Kombination von zwei Gegenständen muss bei einem die Aktion *„Gegenstand abgelegt ⇠⇢“* gewählt werden. Dabei ist darauf zu achten, dass die Aktion mit dem Symbol „⇠⇢“ gewählt wird, damit die Interaktion in beide Richtungen funktioniert.
Im Aktionsfenster werden nun folgende Aktionen eingestellt:
Ausgangs-Objekt 1 (Lighter) von Person entfernen
Ausgangs-Objekt 2 (Fabric) von Person entfernen
Zielobjekt (Fabric) zu Person hinzufügen
Werden beide Ausgangs-Objekte nun aufeinander gezogen verschwinden Sie aus dem Inventar und das neue Zielobjekt erscheint. Auf diese Art wurden alle Interaktionen im Spiel erstellt.

SCREENSHOTS
05 00
INVENTAR

INVENTAR

INVENTAR

INVENTAR

INVENTAR

INVENTAR

FAZIT & AUSBLICK

Die Realisierung dieses Projekts war insgesamt eine große Herausforderung. Von Recherche und Konzepterstellung über Illustrationen, Interfacedesign und Animation ergab sich eine ganze Bandbreite an designrelevanten Aufgaben.

Der fertiggestellte Prototyp des Spiels schafft nun eine gute Basis für die weitere Produktion und Fertigstellung des ganzen Spiels. Anhand des Prototypen können erste Test-Spiele durchgeführt werden, um die Benutzerführung in einer Spielsitutation zu testen und auszuwerten. Fehler oder Mängel im Gameplay können so erkannt und für die weitere Entwicklung optimiert werden.

Die Herausforderung in dieser Dokumentation bestand darin, den kompletten Entwicklungsprozess abzubilden und für interessierte Leser nachvollziehbar zu machen. Dabei mussten viele teils sehr komplexe technische und gestalterische Vorgänge kurz und präzise dargestellt werden. Eine kontinuierliche Überarbeitung und Erweiterung der Inhalte erscheint als lohnenswert, da ein vergleichbarer Leitfaden zur Erstellung eines eigenen Adventures in dieser vollständigen Form meines Wissens nach noch nicht existiert.

QUELLEN

Bücher

- Ollie Johnston und Frank Thomas, The Illusion of Life: Disney Animation
- Richard E. Williams - Animator‘s Survival Kit
- Chris Crawford on Interactive Storytelling (New Riders)
- Tom Bancroft Creating Charakters with personality

Internet

www.freesound.org
de.wikipedia.org
www.visionaire-studio.net
www2.visionaire2d.net/glenfx
www.videohive.com

ASSERTION

Hiermit versichere ich, dass ich die Arbeit selbstständig angefertigt habe und keine anderen, als die angegebenen und bei Zitaten kenntlich gemachten, Quellen und Hilfsmittel benutzt habe.

Volker Zerbe, Mönchengladbach, März 2012

Intermediate Project Hauptthema
© Volker Zerbe, März 2012

Audiovisuelle Medien
Hauptprüfer: Prof. Björn Bartholdy
Nebenprüfer: Prof. Andreas Wrede

Köln International School of Design