

Schon immer hatten die Editoren mancher Spiele eine magische Anziehungskraft auf mich. Die Möglichkeit, sich selbst Missionen, Waffen oder andere nette Spielereien nach seinem eigenen Wunsch zu erstellen, ist meiner Meinung nach ein wahres Geschenk. Bis es jedoch so wird, wie man es sich vorgestellt hat, vergeht einige Zeit und sehr oft folgt Ernüchterung über die Durchsetzbarkeit mancher Projekte. Morrowind ist hierbei eine wundervolle Ausnahme, denn beinahe alles lässt sich den eigenen Wünschen anpassen, was eine unglaubliche Fülle von Möglichkeiten mit sich bringt - aber viele Möglichkeiten bedeuten auch viele Funktionen und oft genug führt dies zu Verwirrung, was ich immer wieder bei meiner Modtätigkeit, und in früherer Zeit auch an meinem eigenen Leib erleb(t)e. In diesem Tutorial möchte ich mein Wissen, das ich während meiner Modder Tätigkeit sammelte, sowie das der Leute, die mir halfen dieses Tutorial zu schreiben mit euch teilen. Es soll ein umfangreiches Kompendium über alle Bereiche des CS darstellen und so viele Fragen und eine Menge Frust unnötig machen.

mfg
madmaxx

Whats New?

In dieser Version sind neben den neuen Namen auch einige Kapitel und Erweiterungen hinzugekommen.

Insgesamt gibt es 11 neue Seiten auf denen nun folgendes zusätzlich besprochen wird:

Neuer Abschnitt: Sauber modden (thx tzo lord of amaya)
Zusatz 'Interiordesign für Anfänger: Snap to Grid / Snap to Angle'
Zusatz 'Troubleshooting': Fehlermeldung beim Laden von Tribunal und Bloodmoon
Neuer Abschnitt: Regions & Region Paint
Neuer Abschnitt: Magic Effects
Neuer Abschnitt: Skills
Neuer Abschnitt: Class
Neuer Abschnitt: Birthsigns
Neuer Abschnitt: Faction

Viel Spaß mit dem neuen alten Tutorial
mfg
madmaxx

'Disclaimer: TES, Morrowind, Daggerfall, Arena, Tribunal, Bloodmoon, the TES Construction Set etc. are property of Bethesda Softworks, a ZeniMax Media company. The author of this manual makes no claims to these names and trademarks. This guide is completely unofficial and in no way am I, nor is Bethesda, responsible for any damages or loss of data, patience or sanity inflicted through the use of this manual. You can freely distribute this document as long as you keep it intact and unmodified.'

TES, Morrowind, Daggerfall, Arena, Tribunal, Bloodmoon, das TES Construction Set etc. sind Eigentum von Bethesda Softworks, eine Firma von ZeniMax Media. Der Autor dieser Anleitung erhebt keinerlei Anspruch auf diese Namen und Markenzeichen. Dieser Guide ist komplett unofficial und weder ich noch Bethesda Softworks sind haftbar für jegliche Schäden oder Datenverlust, die durch Gebrauch diese Anleitung verursacht wurden. Ich gebe keine Garantie auf Richtigkeit der in diesem Tutorial gemachten Aussagen. Sie können dieses Dokument frei verbreiten, solange der Inhalt unverändert bleibt und es nicht zu kommerziellen Zwecken missbraucht wird.

Dieses Tutorial ist Anna gewidmet. Verloren für doch so lange Zeit.



	Seite:
I. Grundlegendes	5
•1.1 Plugins und Masterfiles	5
•1.2 Verzeichnisstruktur	5
•1.3 Grundlegende Begrifflichkeiten	5
•1.4 Nützliche Programme	6
•1.5 Sauber modden	8
II. Übersicht über den Editor	12
•2.1 Die Bedienfelder des Editors	12
•2.2 Das Objektfenster	14
•2.3 Das Cellview Fenster	14
•2.4 Das Renderfenster	15
•2.5 Tastaturkommandos	15
III. Erste Schritte	16
•3.2 Exteriordesign für Anfänger	16
•3.3 Interiordesign für Anfänger	21
•3.4 Feinschliff	27
IV. Neue Objekte	29
•4.1 Waffen	29
•4.2 Bodyparts	30
•4.3 Kleidung & Rüstung	31
•4.4 Sonstiges	32
V. NPC's	34
•5.1 Erstellen eines NPC's	34
•5.2 Service NPC's	35
•5.3 AI	36
•5.3.1 Wander	36
•5.3.2 Travel	36
•5.3.3 Escort	37
•5.3.4 Follow	37
•5.3.5 Activate	37
•5.4 Animationen	37
VI. Dialoge	38
•6.1 Das Dialogue Window	38
•6.2 Topics	39
•6.2.1 Erstellen von Topics	39
•6.2.2 Conditions	41
•6.2.3 Multiple Choice	41

•6.2.4 Dialogue Result	41
•6.2.5 Text Defines	42
•6.3 Greetings	43
•6.4 Voice	43
•6.5 Journal	44
•6.6 Persuasion	44
VII. Exterior design für Fortgeschrittene	45
•7.1 Verwendungsmöglichkeiten von Vertex Paint	45
•7.2 Wälder	46
•7.2.1 Grundlegendes	46
•7.2.2 Licht & Ton	47
•7.3 Tipps&Tricks	48
•7.3.1 Überhänge	48
•7.3.2 Wasserfälle	49
•7.3.3 mehrstufiges Wasser	49
•7.4 Städte	50
•7.5 Regions & Region Paint	
VIII. Interiordesign für Fortgeschrittene	52
•8.1 Layout	52
•8.2 Atmosphäre	53
•8.2.1 Deko	53
•8.2.2 Beleuchtung & Sound	53
•8.3 Tipps&Tricks	54
•8.3.1 Gallerien	54
IX. Rassen	55
•9.1 Das Rassenfenster	55
•9.2 Erstellen einer neuen Rasse	55
•9.2.1 Konzeption	56
•9.2.2 Fähigkeiten und Eigenschaften	56
•9.2.3 Bodyparts	56
X. Dropdown Gameplay & Character	57
•10.1 Game Settings	57
•10.1.1 Das Game Setting Fenster	57
•10.1.2 Anwendungsgebiete	58
•10.2 Magic Effects	58
•10.3 Skills	59
•10.4 Class	60
•10.5 Birthsigns	60
•10.6 Faction	61
XI. Verschiedenes	63
•11.1 Die Morrowind.ini	63
•11.2 Übersetzung von EV-MOD's	67
XII. Troubleshooting	70
•12. Häufig gestellte Fragen	70
XIII. Danksagung	72



1.1: Plugins und Masterfiles



»Was ist eigentlich ein Plugin?«

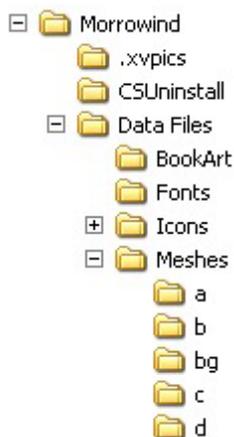
Diese Frage lässt sich relativ einfach klären. Plugins (mit der Dateieendung *.esp*) beinhalten alle Änderungen an dem oder den Masterfiles (*.esm*) oder anderen Plugins, die gemacht wurden. Nehmen wir beispielsweise einmal an, wir würden im Original MW (daher der *Morrowind.esm*) in Pelagiad ein neues Haus hinstellen, so stünde in dem Plugin lediglich die Koordinate und rotation des Hauses sowie welches Haus verwendet wurde.

Allerdings werden in Plugins nicht nur Veränderungen an Masterfiles gespeichert, sondern auch an anderen Plugins, die zu dem Zeitpunkt geladen werden.

Daher ist Vorsicht geboten, wenn man beim Modden mehrere Plugins geladen hat, jedoch kann man dies auch ausnutzen, beispielsweise in Form eines Patches.

Masterfiles hingegen sind oft Total Konversionen oder sehr große Mods, bei denen es sich rentiert, die Eigenschaften der Masterfiles wie beispielsweise keine vervielfältigung, wie sie oft auftritt wenn man mit verändertem Plugin einen Spielstand mit der alten version läd.

1.2: Verzeichnisstruktur



Die Verzeichnisstruktur des Morrowind/Data Files Ordners mag auf den ersten Blick verwirrend und die Buchstabenkennungen mögen kryptisch wirken, aber dem ist nicht so.

Hier einmal eine kurze Erklärung der Ordner:

•**Data Files** - Der Hauptordner. Hier befinden sich Plugins, Masterfiles und BSA Archive

•**BookArt** - Hier befinden sich Bilder, die in Büchern auftauchen

•**Fonts** - Die verschiedenen Schriftarten des Spiels

•**Icons** - Hier werden die kleinen Icons der Objekte gespeichert. Dieser Ordner hat mehrere Unterordner die ich weiter unten beschreiben werde

•**Meshes** - in diesem Ordner befinden sich die Models des Spiels - auch hier gibt es eine Menge Unterordner.

•**Music** - Musik eben.

•**Sound** - Ob Waffeengeräusche oder Begrüßungen, alles ist hier drin.

•**Splash** - Splash Screens sind die Ladebildschirme, die auftauchen wenn man beispielsweise ein Spiel lädt.

•**Textures** - Die Texturen des Spiels, dummerweise ungeordnet. Hier ist nur ein Unterordner von Interesse

Die Unterordner sind wie folgt geordnet:

- **a** - **Armor** - Rüstungen und Schilde
- **b** - **Bodyparts** - Körperteile eben aber auch Einzelteile von Rüstungen
- **c** - **Clothing** - Kleidung
- **d** - **Door** - Grafiken der Türen
- **e** - **Effects** - Magische Effekte
- **f** - **Activators&Furniture** - Knöpfe z.B., sowie die ganzen Einrichtungsgegenstände
- **i** - **Interiors** - Räumlichkeiten und alle anderen Interior Statics
- **l** - **Light** - Fackeln, Kerzen, etc
- **m** - **Misc** - Geld, Dwemergegenstände, etc
- **n** - **Ingredients** - Zutaten
- **o** - **Container** - Pflanzen, Fässer, Kisten, etc
- **r** - **Creature** - Tiere, Monster
- **w** - **Weapon** - Waffen eben
- **x** - **Exterior** - Felsen, Bäume, Exterior Häuser

Im Ordner Sound verhält es sich wie folgt:

- **Cr** - **Creature** - Sounds der Kreaturen
- **Fx** - Effekte wie z.B. Schüsse, Schwerterklirren, Schritte, etc
- **Vo** - **Voice** - Die Stimmen.

Im *Music* Ordner gibt es:

- **Battle** - Musik die bei Kämpfen abgespielt wird
- **Explore** - Musik die abgespielt wird, wenn man normal durch die Gegend läuft
- **Special** - Wie der Name schon sagt besondere Musik wie beispielsweise bei der Chargen

1.3: Grundlegende Begrifflichkeiten

Ich werde in diesem Tutorial einige Begriffe und Abkürzungen benutzen, die man einfach kennen **muss**.

MW - **Morrowind**

CS - **Construction Set**, d.h. der Editor von Morrowind

ID - Eine ID ist ein **einzigartiger** Name, anhand dessen das Spiel die einzelnen Gegenstände unterscheiden kann.

Referenz - Ein Gegenstand kann öfter als einmal im Spiel vorkommen. Dennoch kann es **keine** zwei Gegenstände mit gleicher ID geben. Daher gibt es Referenzen. Das sind Kopien des Original-Gegenstandes, die hinter der ID eine einzigartige Nummer haben z.B. *test_000001*, *test_000002*, etc. Alle diese Gegenstände haben die gleichen Eigenschaften wie das Original.

Ändert man im Editor also einen Wächter, so ändert man auch alle *Referenzen* dieses Wächters im Spiel. Daher ist es so wichtig, **immer (!!!)** neue Gegenstände zu erstellen, es sei denn man will die vorhandenen abändern.

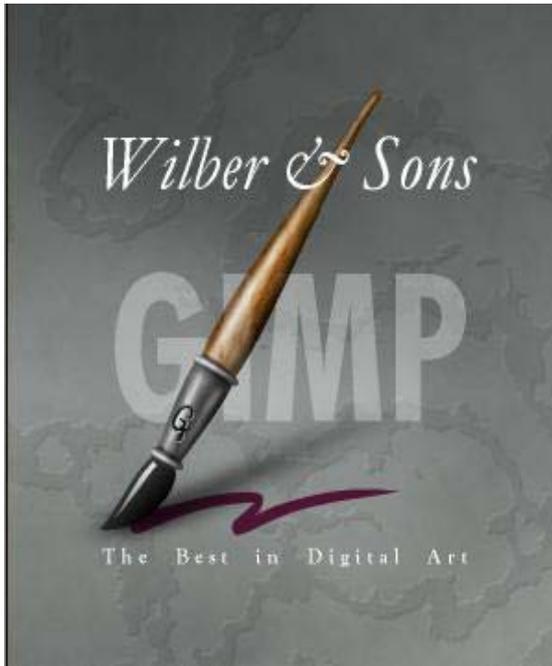
1.4: Nützliche Programme

Früher oder später werdet ihr eine Idee haben, die es nötig macht, auch ausserhalb des CS (Construction Set) zu arbeiten. Womit ihr arbeiten könnt, werde ich in diesem Abschnitt genauer erläutern.

Skinning:

Wenn ihr eine neue Rasse machen wollt, jedoch keine Lust bzw nicht das richtige Programm (3ds max) habt um neue Models zu machen, reicht es oft schon den bestehenden Models eine neue Textur zu verpassen, im Volksmund *umskinnen* genannt.

Hierfür kann ich euch drei Programme ans Herz legen:



Das erste ist das Freeware-Grafik-Tool **GIMP**, welches ursprünglich das »Malprogramm« von Linux war, jedoch für Windows angepasst wurde.

Es beinhaltet eigentlich alle wichtigen Funktionen, die man benötigt, um eine neue Textur zu erstellen.

Beachtet bitte, dass Texturen als **tga** oder **dds** im Data Files/Textures Ordner liegen sollten und eine Größe von **2,4,8,16,32,64,128,256,512,1024** haben müssen.

Dabei müssen sie jedoch nicht quadratisch sein, eine Textur mit der Größe von 128x256 funktioniert ebenso wie eine mit 256x256.

Herunterladen könnt ihr dieses Tool, indem ihr den Links auf <http://www.gimp.de> folgt.

Als nächstes solltet ihr die Textur in das **dds** Format konvertieren, da dieses sehr klein und damit speicherschonend ist. Auf www.rhuarc.de gibt es das Freeware Programm **DDS-View**, das auch als dds speichern kann.

Das dritte und letzte ist schließlich das Programm, mit dem man die Textur auf das Model überträgt. Mein Favorit heißt schlicht **NifTexture** und ist ebenfalls auf rhuarc.de zu haben.

Modeling:

Wenn ihr euch nicht damit zufrieden gebt Objekten bloß eine neue »Haut« zu geben, müsst ihr eben neue machen!

Leider ist dies nicht ganz so einfach, was in dem Model-Format, das Morrowind benutzt, begründet liegt.



Genauer gesagt liegt es an den fehlenden Exportern für *.nif* Dateien.

Bethesda hat zwar Exporter bereitgestellt, jedoch nur für das Profi-Tool 3ds max, welches für die meisten unerschwinglich sein sollte.

Wenn ihr jedoch Glück habt, könnt ihr in nächster Zeit billig 3ds max 4 oder 5 abgreifen, da die Version 6 freigegeben wird.

Glücklicherweise gibt es jedoch auch ein Freeware-Tool von der selben Firma, nur leider keinen Exporter dafür,

so dass man weiterhin auf Leute, die 3dsm haben, angewiesen ist.

Wollt ihr es dennoch versuchen, so könnt ihr euch **gmax** auf der Seite des Herstellers (www.discreet.com) herunterladen...

Sonstiges:

Ein "must have" für jeden Modder sollte **TESAME**, der **The Elder Scrolls Advanced Mod Editor** sein, mit dem man die Änderungen, die eine Mod macht anschauen und ggf. löschen kann.

Auf diese Weise kann und sollte überprüft werden, ob die Mod, die man abliefert, »clean« ist, d.h. nichts ändert, was sie nicht auch ändern soll.

Illadrión war freundlicherweise bereit, mir sein Tutorial über dieses umfangreiche Thema zur Verfügung zu stellen, sobald dieses fertig ist. Herzlichen Dank an dieser Stelle.

Vor kurzem entdeckte ich ein weiteres Tool, mit dem Namen **mwedit**. Es handelt sich dabei um nichts weniger als um einen neuen Editor.

Dieser befindet sich zwar noch in der Beta-Phase (Version 0.5.5), aber was er bis jetzt kann ist um einiges besser als der Original Editor.

Das mag daran liegen dass er von Moddern entwickelt wurde und daher alle Funktionen, die im Original CS vermisst wurden, enthalten sind.



Die Datei nach dem ersten speichern..

jedoch steht hinter der Anzahl (=count) ein Stern ("*"), dies heißt, dass dieses Objekt im Editor verändert wurde (ist auch logisch, ich habe es ja erstellt). Setzen wir dieses Objekt nun auch gleich ein. Wiederum speichern wir ab, um zu sehen wie sich die Dateigröße verändert hat.

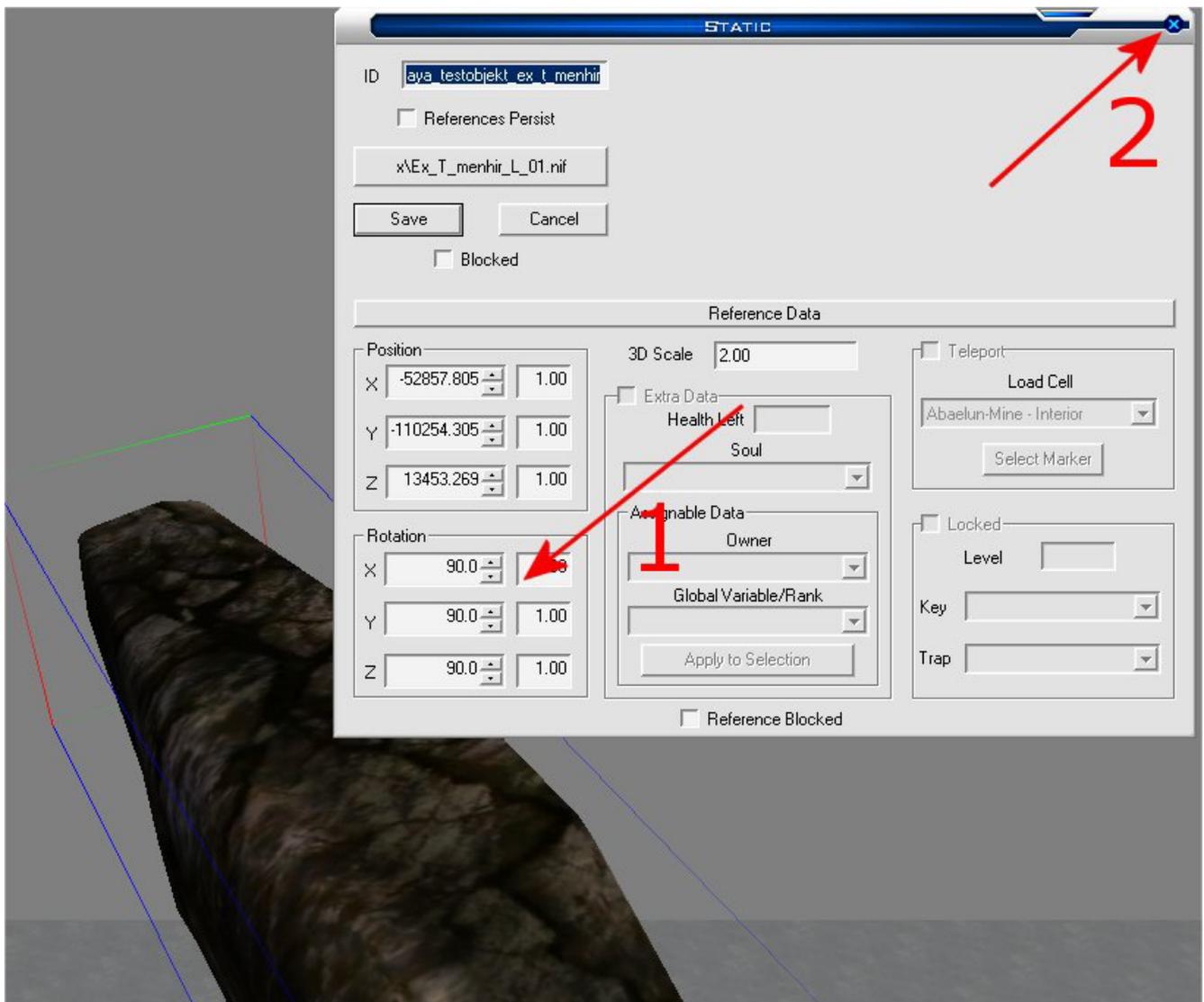


Die Datei nach dem einsetzen des Felsens

Sodele, 4.08 KB oder 4177.92 Bytes (ungerundet), soviel benötigt Morrowind um ein unverändertes PlugIn abzuspeichern (wenn ich dies mit TESAME öffne sehe ich kein verändertes Objekt).

Also dann wollen wir den Versuch starten, ich erstelle nun ein Objekt, welches die selben Eigenschaften hat wie das Objekt "ex_t_menhir_l_01", jedoch eine andere ID, nämlich "amaya_testobjekt_ex_t_menhir", dies hat eine Anzahl von 0,

Nun wiegt das PlugIn schon 4.32 KB, das heisst 4423.68 Bytes, das ergibt 245 Bytes mehr als Vorhin, diese werden durch das neue Static sowie das eingesetzte Objekt benutzt! Wenn wir das Objekt jetzt drehen und vergrössern und sonst wie ändern und dann auf das Kreuz oben rechts klicken um das Fenster des Objekts zu schliessen, sehen wir das die Änderungen am Objekt auch ohne "save" übernommen werden:

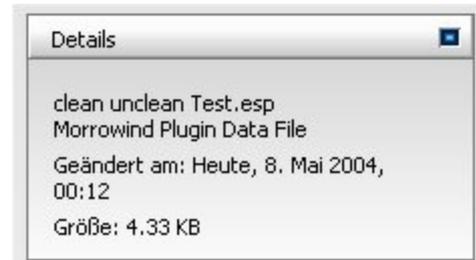


Man sieht im Hintergrund schon das geänderte Objekt obwohl ich das Fenster noch nicht geschlossen habe!



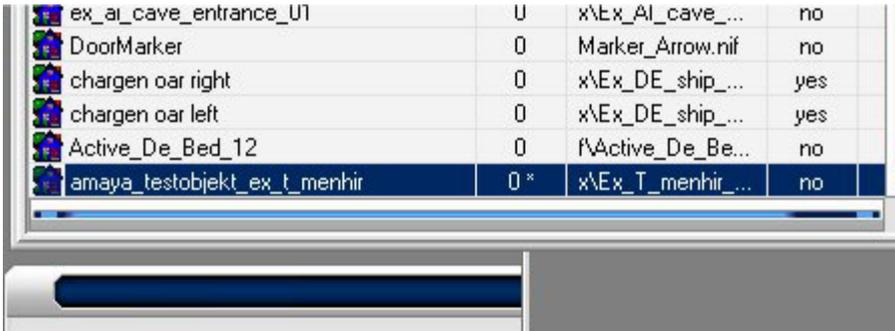
Die Datei hat nun bereits 4,33 KB

Ändern wir das Objekt nocheinmal und drücken "Save" hat sich die Größe nicht geändert.



Auch nach einer weiteren Änderung ändert sich die Größe nicht

Der Mod wurde nicht unclean, weil ich ein neues Objekt mit meinem Präfix erstellt habe. Wenn wir jetzt dieses Objekt wieder löschen, bleibt folgendes zurück:



Wenn ich dieses Eintrag lösche (in der Objects Liste delete drücken) und das PlugIn abspeichere bleibt eine Größe von 4,15KB. Oder 4249.6 Bytes also eine Differenz von 71.68 Bytes.

Jetzt fragt man sich woher diese kommen, schliesslich ist im Spiel das keine Veränderung sichtbar. Hier die Erklärung: Wir haben die Zelle Bitterküse -7,-14 geändert, auch wenn wir das Objekt wieder gelöscht haben ist die Zelle geändert, und so wird es auch gespeichert, siehe hier:

Bitterküste	-8, 1	179
Bitterküste	-8, 2	266
Bitterküste	-8, 4	189
Bitterküste	-7, -14 *	2
Bitterküste	-7, -13	0
Bitterküste	-7, -12	0
Bitterküste	-7, -11	0
Bitterküste	-7, -10	0
Bitterküste	-7, -9	0
Bitterküste	-7, -8	46

Wenn ihr diese Änderung auch noch los werden wollt müsst ihr mit TESAME euer PlugIn laden und die Zelle löschen,

dann hätten wir wieder die Originale Grösse von 4.08 KB und der Mod wäre wieder "clean". Dies ist aber nur der Fall wenn ihr eine Zelle unabsichtlich verändert, die Zellen die ihr bewusst ändert (weil ihr dort ein Objekt hinstellt), müsst ihr natürlich lassen, sonst wäre eure Arbeit ja sinnlos gewesen ;).

So jetzt wollen wir mit dem Original Static arbeiten, und mal aufzeigen was der unterschied zwischen Clean und Unclean ist..

Also nehmen wir das Original unseres Testobjekts (*ex_t_menhir_l_01*) und stellens in die leere Zelle, nun erscheint bei Count der geänderte Wert "9831" (original 9830).

hen jedoch keine Stern, das Programm speichert nur die Änderung in der Zelle und somit ergibt sich eine Größe von 4,24KB.

Diese bleibt auch bestehen wenn wir die Referenz im Renderwindow drehen und **nicht** auf "Save" drücken.

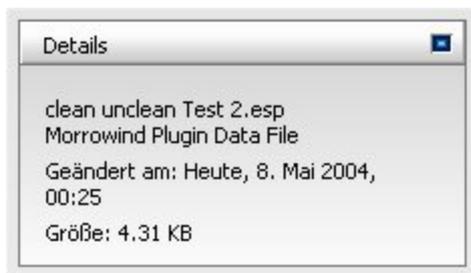
Das PlugIn ist immernoch clean, in der Liste der Objects sehen wir auch kein Stern, das heißt zur Zeit ist nur die Änderung in der Zelle gespeichert.

Wir löschen das Objekt wieder, nun ist wieder nur noch die veränderte Zelle (ohne Objekt) gespeichert, bei unserem vorherigen Test hatten wir an diesem Punkt 4.15 KB Dateigrösse.

Nun wiederholen wir das ganze und drücken diesmal "Save".

Sehen wir und nun das Object Window an, sehen wir, das das Objekt dort mit einem Stern gekennzeichnet ist!

Auch nach dem löschen bleibt dieser Stern erhalten, die Mod ist unclean geworden!



Nach dem löschen haben wir nicht die erwarteten 4,16KB

was dies für einen riesengrossen Mod wie Isle of the death oder Arktwend bedeutet, na ja gute Nacht, das dürft ihr euch selbst vorstellen!

Speichern wir das ganze und sehen uns die Größe an, merken wir, dass diese ganz erheblich von den erwarteten 4,16KB abweicht, die Mod ist unclean..

Wer jetzt denkt blabla 0.15 KB ist ja nichts, stimmt, aber wenn man bedenkt wir haben nur **EINEN** Stein hinein editiert und wieder weggemacht. Es handelt sich um 1 Objekt, jetzt stellt euch das mal mit 1000 Objekten vor, das macht dann 150 KB das sind, das heisst bei einer Mittelgrossen Stadt kann sich eine ganze Menge Müll ansammeln wenn ihr Unclean arbeitet, und

Noch ein Tipp für die Cleanen Leute: Falls es euch immer auf den Geist geht das Kreuzchen zu benutzen beim Schliessen, drückt auf Escape hat die selbe Wirkung, und man gewöhnt sichs schneller an!

Also arbeitet Clean, denn wir wollen noch einen Menge schöne Mods sehen, ohne dass unser Rechner zusammenbricht!

Für Fragen und Anregungen stehe ich gerne bereit, entweder über ICQ/MSN/Mail (siehe Kontakt) oder im Forum ansprechen. Im Forum könnt ihr dies auch mit anderen Moddern besprechen, die noch bedeutend mehr Ahnung haben als ich!

Ich hoffe dieses Tutorial/Bericht/wasauchimmer hat euch gefallen, es war mein Erstes! (Darum verzeiht meine Schreibweise) Ich hoffe ich konnte die Sache einfach erklären, denn Einfach ist diese Sache ganz sicher nicht!

Also bleibt Clean,
Andy

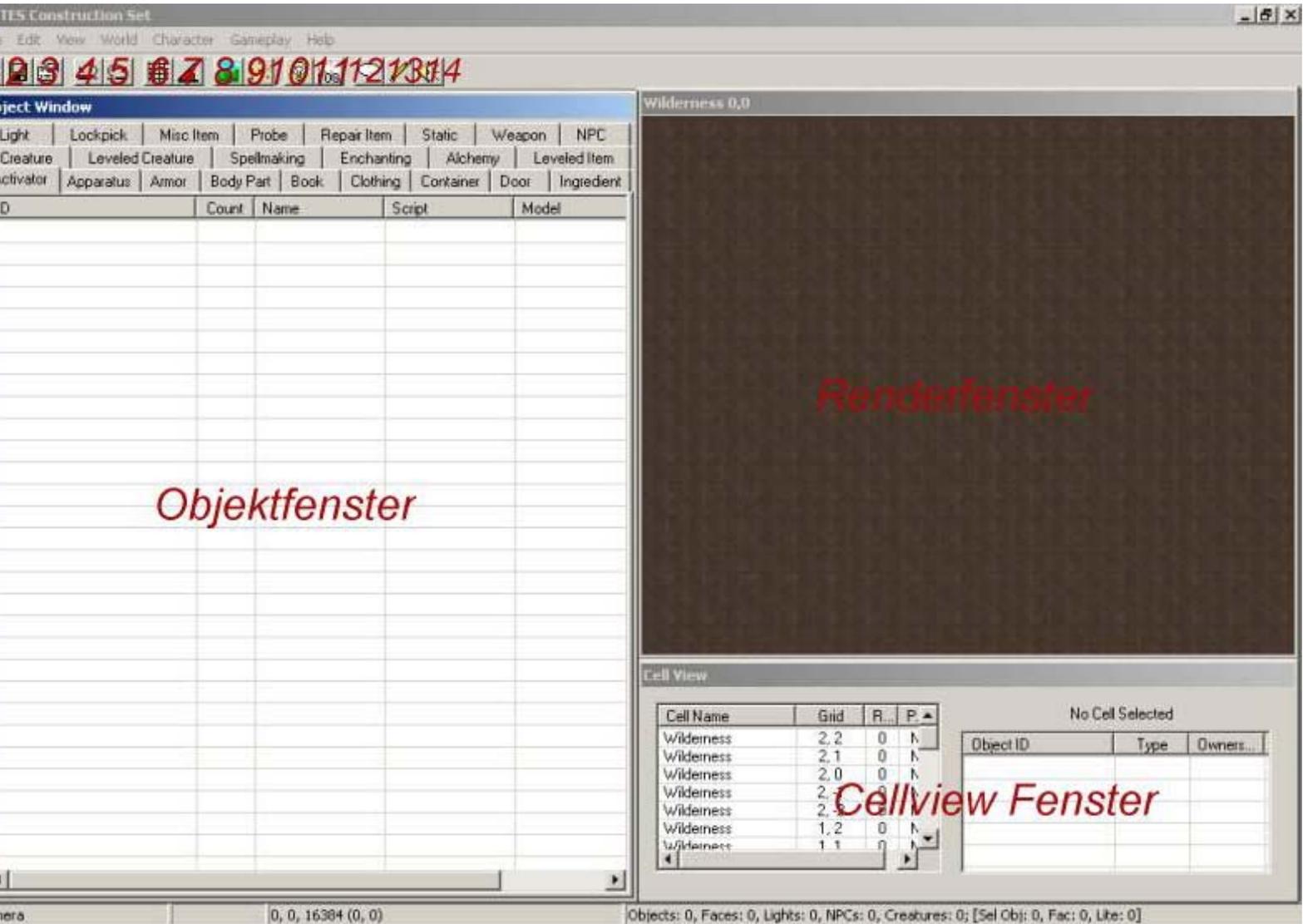


Übersicht

...über den Editor

2.1 Die Bedienfelder des Editors

Sobald der Editor gestartet wurde, begrüßt uns diese Oberfläche:



Objektfenster

Renderfenster

Cellview Fenster

Objektfenster - Hier stehen alle ObjektID's der betreffenden Kategorien, die man oben anwählen kann.

Cellview Fenster - Hier stehen die Zellnamen, ihre Position sowie alle ObjektID's in ihnen

Renderfenster - Hier werden wir einen Großteil der Zeit arbeiten.

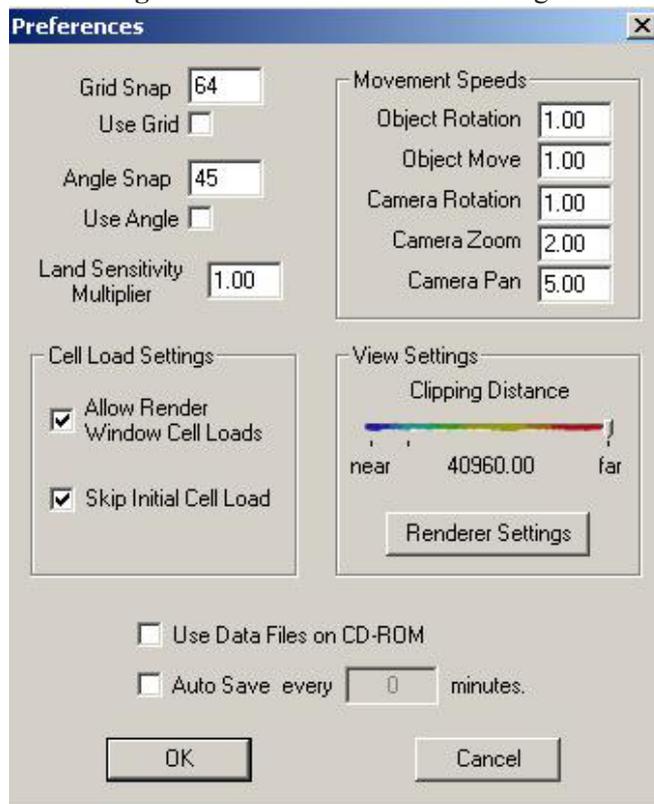
- 1 - Öffnen
- 2 - Speichern
- 3 - Einstellungen
- 4 - Zurück
- 5 - Vor
- 6 - Snap to Grid
- 7 - Snap to Angle
- 8 - Landscape Editing
- 9 - Path Grid
- 10 - Globale Beleuchtung

- 11 - Nebel
- 12 - Dialogue
- 13 - Scripts
- 14 - Sounds

Öffnen - Ist wohl selbsterklärend. Die Plugins werden mit doppelklick selektiert. Das Plugin, an dem gerade gearbeitet wird, sollte als »Active File« gekennzeichnet werden. Wenn kein Active File selektiert ist, kommt eine Meldung, die mit ja bzw yes beantwortet werden sollte.

Speichern - Dazu brauche ich wohl nichts zu sagen.

Einstellungen - Ein Klick hierauf öffnet folgendes Fenster:



Wichtig sind hier lediglich die Einstellungen *Clipping Distance*, *Use Data Files on CD* und *Auto Save*.

Ersteres bestimmt die Sichtweite, ich habe hier mal farbige Makierungen eingezeichnet...

Den roten Bereich sollte man nur für grobes Landscaping benutzen, der gelbe bis grüne ist für normale Arbeiten ausreichend und der blaue ist lediglich für sehr rechenintensive Arbeiten zu benutzen.

Use Data Files on CD bestimmt, ob Meshes, Textures, etc von der Editor CD geladen werden sollen, oder von Festplatte aus dem Ordner *Data Files*.

Ich empfehle **ausdrücklich**, den Ordner *Data Files* von CD auf die Festplatte zu kopieren, da man so schnellere Ladezeiten hat und nicht immer den Ordner wechseln brauch, wenn man zwischen selbsterstellten und MW-eigenen Dateien wechselt.

Vor & Zurück - Wie der Name schon sagt, können hiermit Aktionen rückgängig gemacht, bzw wiederholt werden.

Snap to Grid - Extrem nützlich, da hiermit Objekte automatisch an einem unsichtbaren Raster ausgerichtet werden. Teile von Interiors passen so beispielsweise sofort 100%ig aneinander.

Ich benutze diese Funktion jedoch eher weniger, aus Gründen die später noch näher erläutert werden.

Snap to Angle - Ähnlich wie Snap to Grid, jedoch rastet es hier alle 45° ein. Ebenfalls recht nützlich.

Landscape Editing - Ein Klick hierauf startet den Landscape-Editing-Modus der später noch näher beschrieben wird.

Path Grid - wird erst später wichtig.

Globale Beleuchtung - Schaltet eine weitere, Super Helle Lampe hinzu, die unabhängig von der regionalen Beleuchtung alles 100%ig ausleuchtet - nützlich wenn man dunkle Dungeons baut.

Nebel - schaltet Nebel hinzu - in meinen Augen sinnlos.

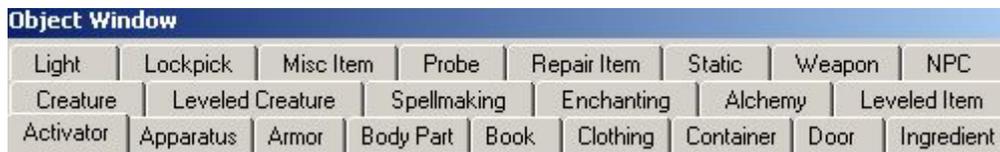
Dialogue - Öffnet das Dialogue Fenster.

Scripts - Öffnet das Scripting Window.

Sound - Öffnet das Sound Window.

2.2: Das Objektfenster

Wollen wir uns das Objektfenster näher ansehen, sollten wir erst einmal die *Morrowind.esm* einladen. Einfach auf öffnen, doppelt auf die *Morrowind.esm* klicken und mit OK bestätigen. Das ehemals leere Fenster ist nun randvoll und wir können uns die einzelnen Kateireiter ansehen.



Object Window								
Light	Lockpick	Misc Item	Probe	Repair Item	Static	Weapon	NPC	
Creature	Leveled Creature		Spellmaking	Enchanting	Alchemy	Leveled Item		
Activator	Apparatus	Armor	Body Part	Book	Clothing	Container	Door	Ingredient

Wenngleich die Bezeichnungen größtenteils selbsterklärend sind, werde ich zu einigen näher Stellung nehmen.

Light - Hier befinden sich (für den Spieler) unsichtbare Lichter, aber auch Kerzen, Kronleuchter, etc.

Misc Item - Alles was nirgendwo anders herein passt, wie Dwemerartefakte beispielsweise.

Static - Alle Gegenstände auf oder in Morrowind, durch die man nicht hindurchlaufen kann, befinden sich hier. Dazu gehören Bäume, Steine, Zäune, Brücken, Rohre, Interior-Bausteine, Exteriorhäuser, Fenster, Mauern, etc...

Leveled Creature - Da man nicht voraussagen kann, mit welchem lvl der Spieler das Plugin spielt, ist es manchmal ratsam, anstelle von 'normalen' Kreaturen leveled Creatures zu setzen, sodass je nach Spielerlvl unterschiedliche Gegner auftauchen und der Spieler jederzeit gefordert wird.

Spellmaking - Alle Zaubersprüche, die irgendwie im Spiel vorkommen - und sei es nur die Wiederherstellung an Schreinen - befinden sich hier. Auch Eigenschaften und Fähigkeiten von Rassen werden hier gespeichert.

Enchanting - Hier befinden sich die Verzauberungen, die es im Spiel gibt.

Lvld. Item - Ähnlich wie lvld. Creature passen sich diese Gegenstände dem Spielerlevel an, sodass in einer Kiste einmal 10 und einmal 1000 Gold liegen können.

Activator - Activators sind beispielsweise Schalter, aber auch die Vivec-Wasserfälle oder Lava gehören dazu.

2.3: Das Cellview-Fenster

Ist die Masterfile geladen, sollten in der linken Spalte sämtliche Zellen von Morrowind aufgelistet sein, standartmäßig alphabetisch geordnet.

Hier sieht man auch einen großen Unterschied zwischen In- und Exterior Zellen.

Während Erstere immer eine Eindeutige ID haben müssen, Beispielsweise »*Vivec, Arena, Versteckter Bereich*« können Exterior Zellen vollkommen gleich heißen, ohne das MW meckert.

Markieren wir nun eine Zelle, wie beispielsweise Balmora, so werden auf der rechten Seite sämtliche Objekte in dieser Zelle angezeigt.

Mit einem Doppelklick hierauf springt die Kamera zu dem betreffenden Objekt, was nützlich sein kann, wenn man ein kleineres Objekt nichtmehr wieder findet.

Doppelklickt man jedoch auf eine Zell-ID springt die Kamera in die Vogelperspektive über die entsprechenden Zelle.

2.4: Das Renderfenster

Das Renderfenster ist den Großteil unserer Zeit unser Hauptarbeitsplatz.

Durch das Renderfenster platzieren wir neue Objekte und NPC's im Spiel, bauen Wälder oder erschaffen Städte.

Alle Angaben am unteren Rand des CS beziehen sich auf das Renderfenster.

Von Links nach Rechts steht dort: ID des angewählten Objekts, Art des Objekts, Position,

Gesamtzahl aller Objekte in der Zelle, der Faces, Lights, NPC's und Creatures.

2.5: Tastaturkommandos

Folgende Auflistung der Tastaturkommandos wird euch beim modden sicherlich behilflich sein.

- **T** Zentriert auf das aktuell gewählte Objekt (in der Draufsicht = Topansicht)
- **C** Zentriert auf das aktuell gewählte Objekt (in der isometrischen Ansicht), Center
- **D** Verwirft die Auswahl (Rotation um Kameramittelpunkt
- möglich)
- **SHIFT** Durch zusätzliche Mausbewegung rotiert die Kamera um das gewählte Objekt. Ist kein Objekt gewählt, ist der Drehpunkt die Mitte der Kamera.
- **SPACE** Durch zusätzliche Mausbewegung wird die Kamera nach oben/unten oder links/rechts bewegt. Alternativ zu SPACE kann auch die mittlere Maustaste benutzt werden.
- **V** Durch die Mausbewegung wird die Kamera nach vorne/hinten bewegt. Alternativ kann man mit dem Mausrad zoomen.
- **Y** Erlaubt das Verschieben eines Objekts entlang der x-Achse.
- **X** Erlaubt das Verschieben eines Objekts entlang der y-Achse.
- **Z** Erlaubt das Verschieben eines Objekts entlang der Z-Achse.
- **F** (=Fall=fallenlassen) Verschiebt ein Objekt solange entlang der Z-Achse, bis es auf die Oberseite eines anderen Objektes trifft (praktisch, um Gegenstände auf Boden, Tischen und Regalen zu verteilen)
- **W** Schaltet zwischen der Wireframe- und gerenderten Ansicht um
- **F4** Zeigt die Kollisionsflächen an
- **L** Zeigt die Radien der Lichtquellen an
- **A** Schaltet eine gleichmässige Lichtquelle hinzu (Nützlich bei dunklen Dungeons oder wenn noch keine Lichtquellen vorhanden), hat keinen Einfluss auf das Erscheinen der Zelle im Spiel.
- **M** Schaltet die Anzeige von Markern (Tür-Teleporter, Lichter, etc.) ab und wieder an.
- **H** Schaltet den Landschaftseditor ein
- **strg + C** (=Copy) Momentan markierte Objekte in die Zwischenablage kopieren. Dabei bleibt die Anordnung der Objekte zueinander erhalten, und man kann auch von einer Zelle in eine andere kopieren.
- **strg + V** (:Paste) Fügt die in der Zwischenablage gespeicherten Objekte in die momentan aktive Zelle ein.
- **strg + X** (=Cut) Entfernt die momentan markierten Objekte von ihrer momentanen Position und fügt sie in die Zwischenablage ein. Danach kann man sie an jedem beliebigen Ort mittels strg + v wieder einfügen.
- **strg + D** (=Double) verdoppelt die momentan markierten Objekte an genau der Stelle, an der sie sich bereits befinden. Sehr praktisch um 'Klone' von Objekte dicht bei einander zu positionieren, einfach verdoppeln und die Kopie etwas verschieben.
- **strg + Shift + V** (=Paste in Place) Fügt das in der Zwischenablage befindliche Objekt an der Stelle ein, an der es kopiert wurde (Raum-Koordinaten). Besonders praktisch, falls man exakte Kopien von Räumen machen will.
- **strg + s** (=Save) Speichert das aktive PlugIn unter dem bekannten Namen, oder öffnet den 'Speicher als...'-Dialog, falls kein PI aktiv sein sollte.

- **B** Blendet die Zellengrenzen ein
 - **F1** Öffnet die (englische) Hilfe des CS, in dem sehr viele Fragen, die hier im Forum gestellt wurden/ werden ganz schnell beantworten könnte... ;) (Gemeint ist das offz. Editor Forum Anmk. d. Verfassers)
 - **strg + Z** (=Rückgängig) Macht pro 1mal drücken einen der letzten Arbeitsschritte rückgängig. Dabei kann man afaik beliebig viele Rückschritte vornehmen.
 - **strg + Y** (=Wiederholen) Wiederholt die zuletzt mit strg + z rückgängig gemacht Aktion.
 - **strg + F** (= Find) Öffnet die CS-interne ID-Suche
 - **F3** (=Find next) Nächste Instanz des gesuchten Objektes anzeigen
 - **F2** (=Find previous) Vorhergehende Instanz des gesuchten Objektes anzeigen
 - **S** (=shrink ?) Markiertes Objekt kann mit Mausbewegung verkleinert und vergrößert werden
- Tastaturbelegung übernommen von Iladrions Editor FAQ 1.4 (by Jadzia_13 & iladrion).

Erste Schritte

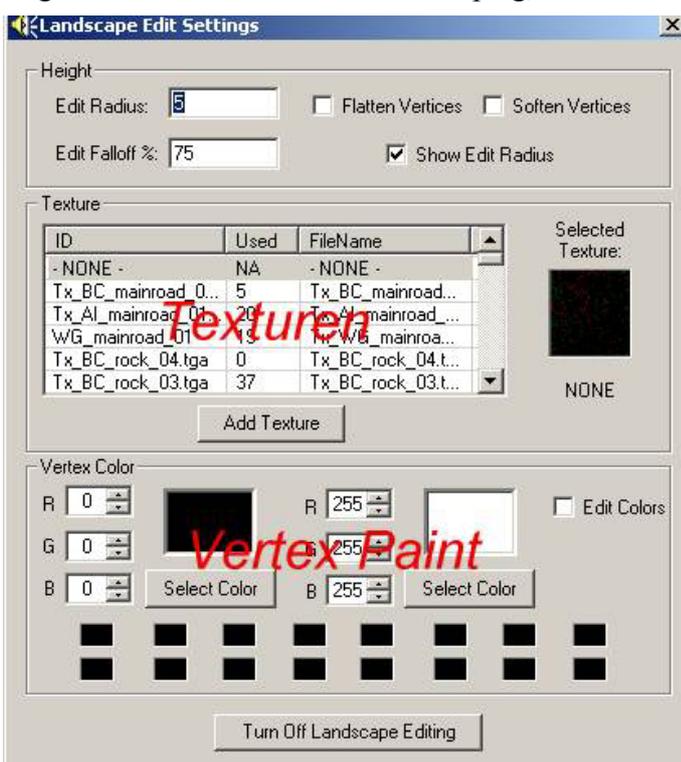
In diesem Kapitel wollen wir eine Insel mit einem kleinen - oder großen Häuschen für unseren Char erschaffen.

Dazu werden wir grundlegende Exterior und Interiordesingtechniken anwenden, um unser Ziel zu erreichen.

Lange Rede, kurzer Sinn, fangen wir direkt an mit:

3.1: Exteriordesign für Anfänger

Exteriordesign heißt für mich, Gelände zu modellieren, texturieren und mit Objekten zu bestücken. Beginnen werden wir mit dem Landscaping, also der Geländemodellierung.



Um in den Landscaping-Modus zu gelangen drücken wir die Taste 'H' oder das Icon im Editor.

Daraufhin sehen wir dieses Fenster sowie einen roten Kreis im Renderfenster.

Der Kreis ist der *Edit Radius* und kann bis auf 30 erhöht werden.

Bewegen wir ihn einmal auf eine Stelle in der Landschaft, halten ihn gedrückt und bewegen die Maus nach oben oder unten, so sehen wir, das das Land abgesenkt bzw. angehoben wird.

Desweiteren gibt es noch die beiden Felder *Flatten Vertices* sowie *Soften Vertices*.

Mit diesen werden wir uns gleich näher beschäftigen.

Unter den Einstellungen ist die Texturen-Liste. Wählen wir einmal eine dieser Texturen an, so wird sie rechts angezeigt.

Wir können die Textur aufbringen, indem wir die rechte Maustaste gedrückt halten und die Maus über die Landschaft bewegen.

Nun wird es Zeit, das Landscaping einmal auszuprobieren.

Ich gehe davon aus, dass die *Morrowind.esm* fertig geladen ist.

Suchen wir uns im Cellview Fenster nun *Seyda Neen -2,-10* und klicken doppelt darauf.

Mit den Cursortasten können wir uns schnell im Renderfenster bewegen, genauer gesagt um eine halbe Cell pro Klick.

Klicken wir einmal Links und einmal unten, so befinden wir uns in der Zelle *Bitterküste -3,-10* bzw. *Bitter Coast -3,-10*.

Zoomen wir als nächstes etwas näher heran und gehen gleich darauf in den Landscape Modus.

Mit einem Edit Radius von 5 erhöhen wir nun etwas Land bis knapp über die Wasseroberfläche und vergrößern das ganze dann mittels Flatten Vertices.

Die Felsen, die im Weg sind makieren wir und löschen sie mit Entf.

(Hinweis: Solange man sich im Landscaping Modus befindet, ist es nicht möglich Objekte zu makieren.)

Das ganze sollte dann ungefähr so aussehen wie auf dem Bild.



Diese kleine Insel ist noch recht trist und unförmig.

Mit einigen Hügeln könnten wir sie wesentlich aufwerten.

Erhöhen wir also das Landscape insgesamt etwas und an einigen Stellen besonders.

Um organisch wirkende Hügel hinzubekommen erhöhen wir das Landscape etwas, machen eine gerade Fläche mit Flatten Vertices und gehen dann mit Soften Vertices und einem großen Edit Radius über alles rüber.

Besonders an der Küste müssen wir Soften Vertices einsetzen damit es nicht allzu steil abfällt...

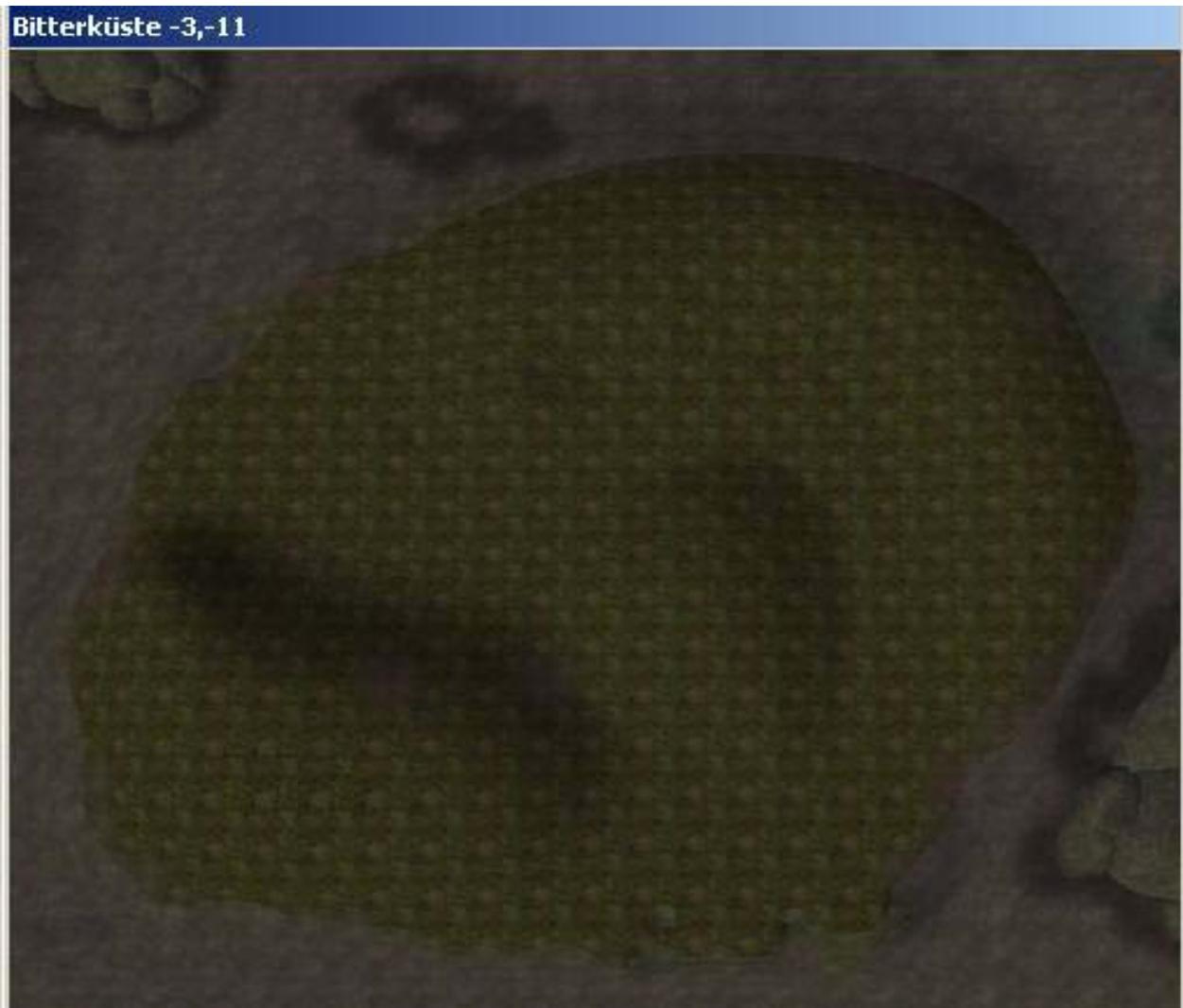
Zum Schluss wird das ganze noch texturiert, z.B. mit *TX_BC_Gras.tga*

Euch wird sicher aufgefallen sein, dass einige schwarze Stellen auf dem Landscape sind.

Diese sind mittels *Vertex Paint* auf der Landschaft aufgemalt und wir werden uns in 7.1 näher damit auseinandersetzen.

Ersteinmal reicht es zu wissen, das man sie ‘ausradieren’ kann indem man unten ein Häkchen bei *Vertex Paint* macht und dann mit der rechten Maustaste über die betreffenden Stellen fährt...

Das ganze sieht nun in etwa so aus:



Schon ganz nett, aber immernoch trist.

Was uns fehlt, sind Objekte wie z.B. Pilze oder Bäume.

Diese befinden sich im Object-Window unter dem Reiter *Static* und beginnen mit dem Prefix *flora_* bzw. *terrain_*

Suchen wir nun das Objekt *flora_BC_tree_01*, fahren sie mit dem Mauszeiger darüber, halten sie die linke Maustaste gedrückt und ziehen sie die Maus ins Renderfenster...

E voilà, ein wunderschöner Baum ;)

Um diesen zu bewegen, makieren wir ihn und halten die linke Maustaste gedrückt während wir die Maus bewegen.

Rotiert wird er mit der rechten Maustaste.

Die andere Möglichkeit besteht darin, doppelt auf den Baum zu klicken und ihn mittels Tastatureingabe zu bewegen bzw rotieren. Wir werden uns im nächsten Abschnitt noch genauer damit beschäftigen, bis dahin reicht die Bewegung per Maus aus.

Nun, drücken wir einmal die Taste *F*, so ‘fällt’ der Baum auf das Landscape.

Jedoch befindet sich die Wurzel dann immernoch oberirdisch, also muss der Baum per Hand noch etwas nach unten verschoben werden, dazu drücken wir *Z* und bewegen die Maus, während die linke Maustaste gedrückt ist.

Na also, war doch garnicht so schwer, oder?

Nun wird es Zeit das erste mal zu speichern, nennen wir das Plugin "My_Plugin".

Sollte das CS nun abstürzen muss beim neustart die *Morrowind.esm* geladen werden sowie *My_plugin.esp* als Active File gewählt werden.

Das ist wichtig, da sonst die Änderungen die wir machen nicht in *My_Plugin* gespeichert werden sondern in einer anderen *esp*

Dekorieren wir unsere Insel nun nach Herzenslust, hierfür bieten sich alle Statics mit den Präfixen *Flora_* sowie *terrain_* an, wobei bei letzterem das Suffix *_bc* (Bitter Coast) vorhanden sein sollte.

Das kann einige Zeit in Anspruch nehmen und zum Schluss sollte es in etwa so aussehen:



Wie ihr seht habe ich eine recht felsige Küste und einen dichten Wald geschaffen aber natürlich ist alles möglich, Grenzen setzt euch nur eure Phantasie, die Zeit die ihr investiert sowie die Leistung eures PCs ;).

So, schön und gut aber es fehlt immernoch eine Menge.

Zum einen muss die Insel noch passend texturiert werden, denn wo wächst das Gras schon ins Meer hinein?

Zum anderen müssen wir dem Landscape den letzten Feinschliff verpassen.

Dazu nehmen wir einen *Edit Radius* der Größe 1 und passen das ganze so an, das es realistisch wirkt.

In meinem Fall wären das steilere Klippen Richtung SO und weiche Unebenheiten im Wald...

Oft ist es dabei hilfreich mit **W** in den Wireframe-Modus zu schalten.

Anschließend texturieren wir. Sand an der Küste, an den Klippen Stein und auch ansonsten etwas mehr Abwechslung.



Ist doch schon ganz ansehnlich, oder?

Nun fehlt noch eine Kleinigkeit, die gleichzeitig der Übergang zum nächsten Abschnitt ist: Ein Hausexterior.

Diese finden wir unter Static und haben das Präfix *Ex_xyz*.

Sucht euch was Schnuckeliges raus und setzt es an einen passenden Platz auf eure Insel.

In meinem Fall benutze ich eine Mischung aus *ex_imp_keep01* sowie *ex_imp_guardtower01* beides auf 2.0 skaliert.



Bevor wir nun fortfahren sollten wir das ganze einmal speichern und Ingame testen um Fehler zu erkennen und zu entfernen.

Bei Tests ist zu beachten, NICHT zu save während das Plugin geladen hat, ansonsten muss man es einmal deaktivieren, das Spiel laden, Fehlermeldungen wegstreichen nochmal speichern, beenden, Plug wieder aktivieren und weiter testen, da ansonsten die Objekte geduped (also zweimal vorhanden) sind.

Also am besten einen sauberen Spielstand anlegen bevor man das Plug aktiviert.

Auch hilft es die Zelle umzubennen z.B. 'Test' und sich dann mittels dem Konsolenbefehl *coc test* hinzubeamen.

3.2 Interiordesign für Anfänger

Nachdem wir eine Insel erschaffen haben geht es jetzt darum, die Innenräume zu machen.

Dazu klicken wir auf *World -> Interior Cell -> New*. Wir nennen sie »Mycell«.

Wechseln wir nun zu *Mycell* und wir sehen... nichts.

Nun, das sollten wir schnell ändern ;-).

Da ich eine Burg in die Gegend gesetzt habe sollte ich den *Imperial-Style* benutzen, der die Abkürzung *imp* trägt.

Die Interiorbausteine, die wir benutzen werden tragen die Präfixe *in_imp* also *Interior_imperial*.

Ich fange meistens bei der Tür an, oder, in unserem Fall mit der Eingangshalle.

Platzieren wir also *in_impbig_4way_01*. Nun kommt der große Unterschied zum Exteriordesign.

Während wir 'draußen' mit der Hand arbeiten können, ist es in Innenräumen wichtig, das alle Teile 100%ig passen, um keine unschönen Lücken entstehen zu lassen.

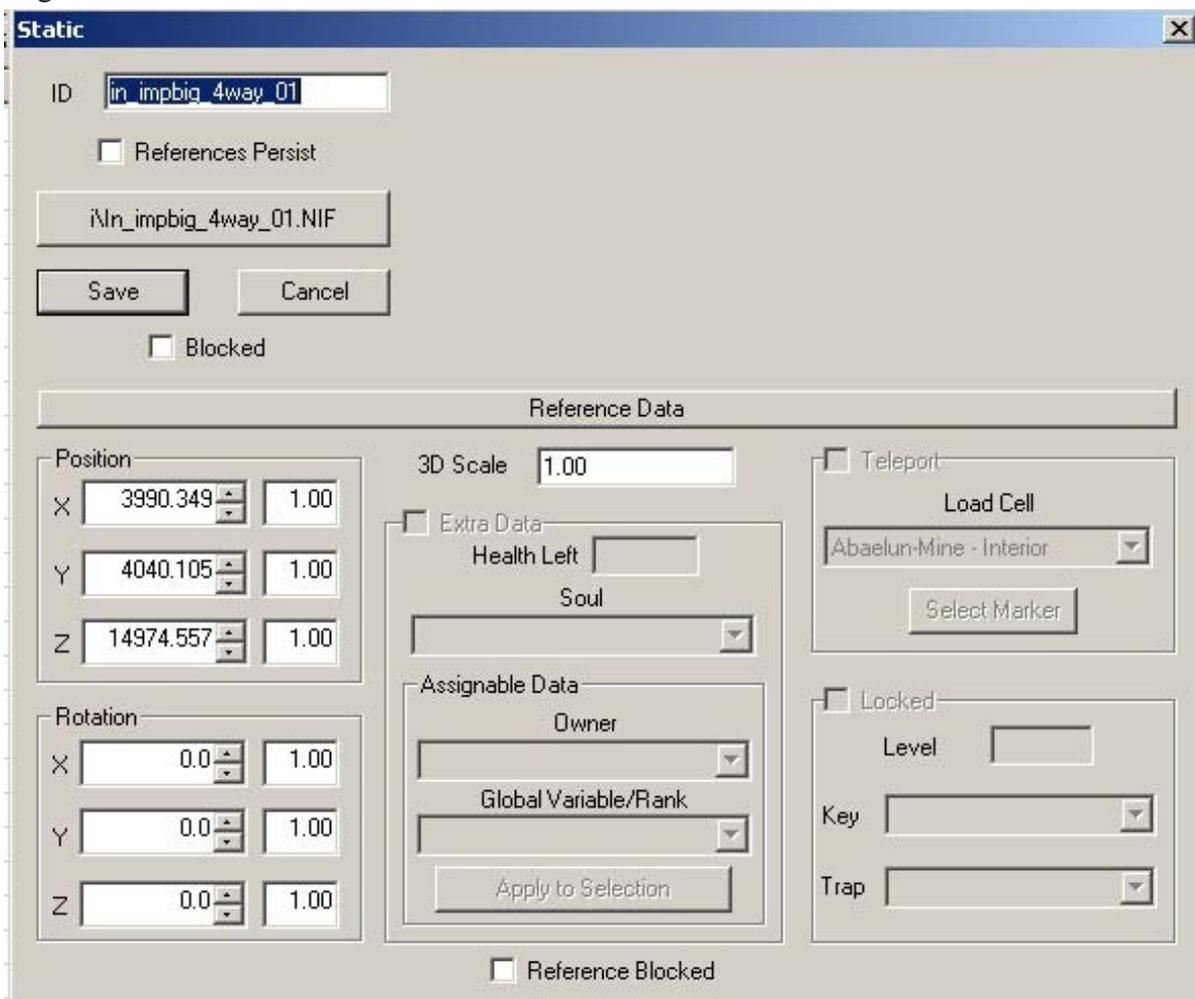
Um dies zu erreichen gibt es viele Möglichkeiten, aber ich schwöre nach wie vor auf die Tastatureingabe.

Eine andere Möglichkeit wäre, *Snap to Grid* zu aktivieren, aber das hat den Nachteil das die Koordinaten oft Nachkommastellen haben. Allerdings ist es gerade bei mehreren Stockwerken wichtig, die Einzelteile 'blind' setzen zu können.

Nundenn, probiert einfach beides aus dann entscheidet selbst was ihr besser findet, ich werde beides im Nachhinein kurz erläutern.

Also weiter.

Im Renderfenster doppelklicken wir nun auf *in_impbig_4way_01* und sehen uns daraufhin mit folgendem Fenster konfrontiert:



In diesem Abschnitt wichtig ist eigentlich nur *Position*, *Rotation* und unter Umständen *3D Scale*. Zu den anderen Schaltflächen werde ich später kommen.

3.2.1: Tastatureingabe:

Geben wir als Koordinaten 0,0,0 ein.

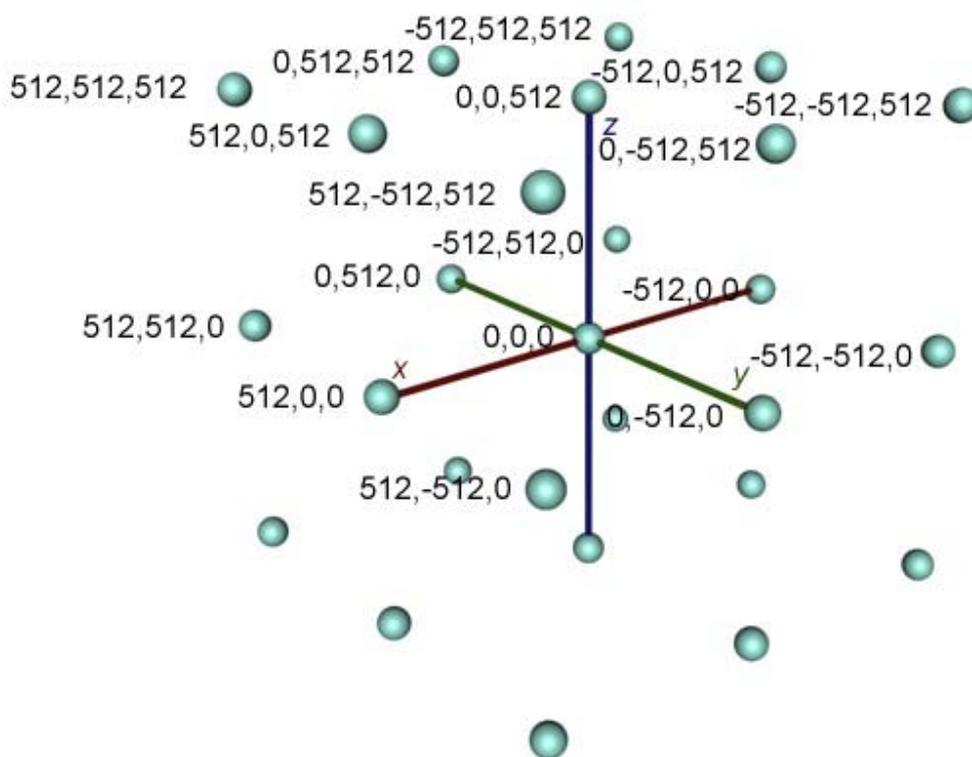
Daraufhin wird es aus dem Sichtfeld verschwinden, holen wir es mit einem Doppelklick auf den Eintrag im Cellview Fenster wieder in Sicht.

Eine Eingangshalle besteht aber nicht aus nur einem Teil sondern aus vielen, aneinandergereihten.

Platzieren wir also noch eine *in_impbig_4way_01* (oder makieren die bereits bestehende und drücken **Strg+D**).

Die nächste kommt auf 0,512,0.

Um euch das zu verdeutlichen habe ich eine Illustration vorbereitet (Ich hoffe ihr habt räumliches



So das sieht erstmal sehr verwirrend aus.

In der Mitte ist 0,0,0, also das Teil, welches wir als erstes platziert haben.

Die Punkte sind die jeweils nächstgelegenen Interiorbausteine.

Der Abstand beträgt deshalb 512, weil es sich um *impbig* handelt - bei *impsmall* beträgt der Abstand die Hälfte.

Die Schlaumeier unter euch werden feststellen das die Achsen net Stimmen, bei x müsste links der neg. Bereich sein, aber das ist net weiter schlimm solange euch klargeworden ist was ich zum Ausdruck bringen wollte.

Wenn nicht, auch nicht schlimm, machen wir weiter.

Duplizieren wir *in_impbig_4way_01* noch ein letztes mal und setzen ihn auf 0,1024,0.

Nun haben wir eine ausreichend lange Eingangshalle.

Am Ende der Halle kommt eine Wand mit der ID *in_impbig_wall_01* und zwar welche Koords.?

Genau! 0,1024,0.

3.2.2: Snap to Grid & Snap to Angle

Nichts einfacher als das ;-). Aktiviert einfach “*Snap to Grid*” sowie “*Snap to Angle*” ..

Wenn ihr nun ein Objekt verschiebt, so werdet ihr merken, das es sich nicht “flüssig” bewegt, sondern ruckartig an einem internen unsichtbaren Raster.

Auf diese Weise kann man sehr schnell große Gebäude bauen, da man nicht lange rechnen und nachjustieren muss.

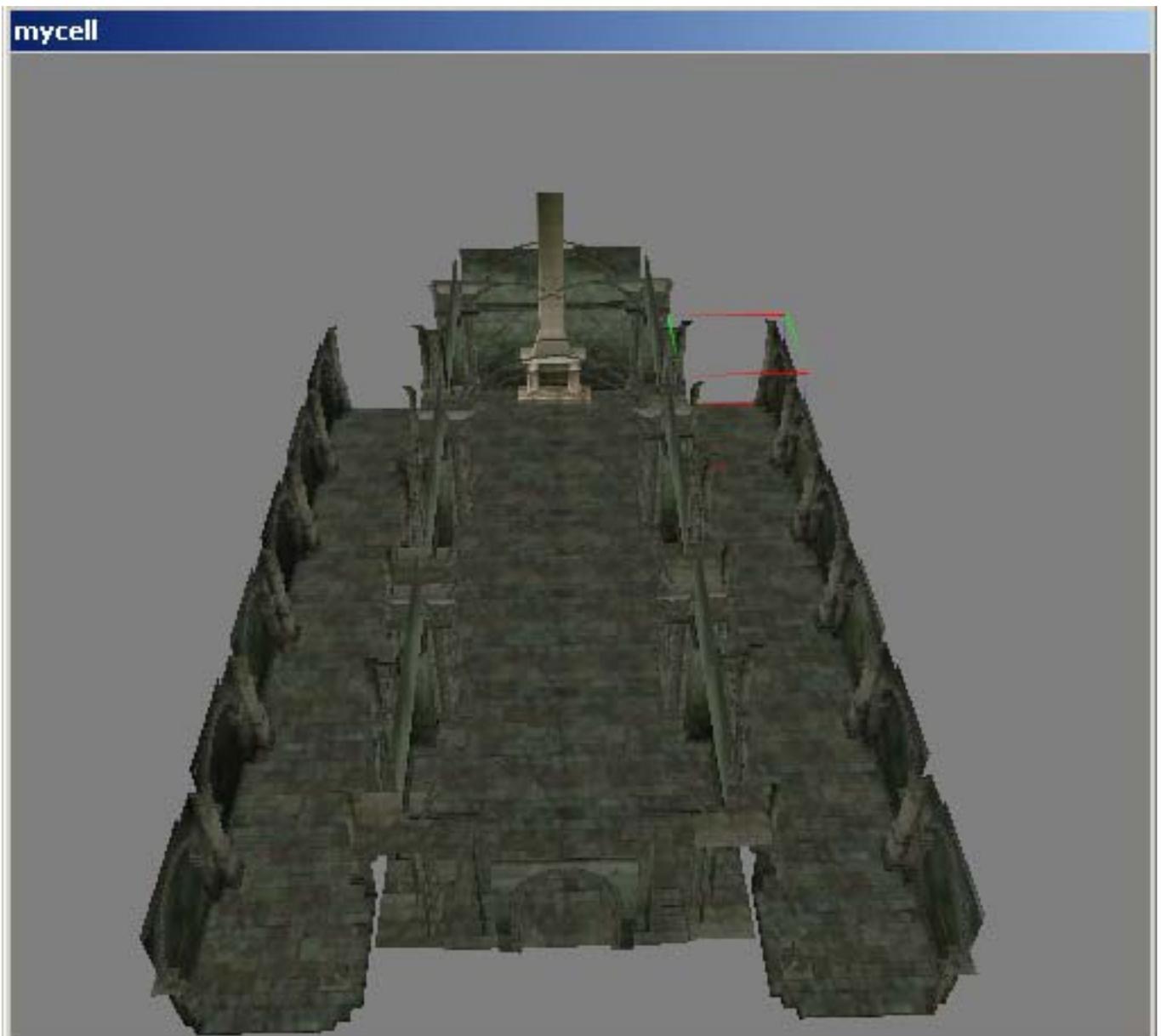
Die Größe des Gitters kann man unter *Preferences* unter den Punkten “*Angle Snap*” sowie “*Grid Snap*” einstellen.

Dabei sollten jedoch unbedingt immer Zweierpotenzen gewahrt bleiben!

Wie auch immer ihr es bis hierhin gemacht habt wir dekorieren das ganze nun mit einem Kamin, der auf den wunderschönen Namen *in_imp_fireplace_grand* hört.

An alle offenen Seiten kommen nun *in_impbig_blend*, um einen Übergang zu den kleinen *in_impsmall* zu bekommen, die nun beidseitig ‘angebaut’ werden.

Das Ganze sollte dann in etwa so aussehen:



Die Seitenteile bestehen, von unten nach oben aus: *in_impsmall_endcap_01*, *in_impsmall_3way_01* sowie *in_impsmall_hall_01*.

Nun, die Grundstruktur steht, nun geht es an die Dekoration und - sehr wichtig - das Licht. Der Screen oben wurde mit zugeschaltetem globalen Licht erzeugt, ohne ist es wesentlich dunkeler. Welche Beleuchtung wäre also angebracht?

Entweder Kronleuchter oder Fackeln... nicht zu vergessen das Kaminfeuer...

Hier jedoch beginnen wir bereits nach *VIII. Interiordesign für Fortgeschrittene* anzudriften, indem ich der Beleuchtung einen eigenen Abschnitt gewidmet habe.

Bleiben wir also erstmal bei den Grundlagen und platzieren einfach einige Kronleuchter (*light_com_chandelier*) sowie Fackeln (*light_com_torch*). Für das Kaminfeuer nehmen wir *light_fire_nosmoke*, welches wir auf 1.2 skalieren.

Auch sollten wir nicht *furn_log* vergessen, welches unter Static zu finden ist.

Habt ihr die Lichter und Holzscheite an ihren Platz gebracht, solltet ihr ersteinmal speichern.

Dann deaktiviert die *Globale Beleuchtung* und seht euch an ob ihr mit der Ausleuchtung zufriedener seid.

Bei mir sieht das dann so aus:



Machen wir das ganze nun noch etwas gemütlicher, indem wir ein paar (Wand-)Teppiche sowie andere Möbel platzieren.

Sämtliche Einrichtung hört auf den Präfix *furn_* für *furniture*.

Auf dem oberen Bild habe ich bereits *furn_imp_sconce_01* (Fackelhalter) platziert.



Hier seht ihr die fast fertig ausgestattete Eingangshalle... Dem aufmerksamen Beobachter wird nicht entgehen, das ich einige Waffen an die Wand gehängt habe, was die Atmosphäre unterstreicht. Nun könnt ihr nach dem selben Prinzip noch weitere Räume bauen, vielleicht einen Weinkeller oder Kerker, ein Schlafgemach und einen Lagerraum. Oder eine kleine Arena. Oder einen Zoo. Auch hier sind eurer Phantasie keine Grenzen gesetzt. Nundenn, es gibt noch ein nicht unwesentliches Problem: Man kann die Burg nicht betreten, da es keine Verbindung zur Aussenwelt gibt...

Um diese zu schaffen, brauchen wir ersteinmal eine Tür.

Diese finden wir unter dem Katereiter *Door*. Dabei ist jedoch zu bachten, das es zwei Arten von Türen gibt: reine *Teleporttüren* und *Interiortüren*.

Teleporttüren wurden nur von einer Seite modelliert, um Polygone und damit Performance zu sparen. Das macht sie jedoch auch nur dafür benutzbar - als Türen in Innenräumen sind sie unbrauchbar.

Interiortüren sind von beiden Seiten modelliert und dafür geschaffen, in Innenräumen zwischen zwei Räumen benutzt zu werden.

In unserem Fall benötigen wir ersteres, genauergesagt *Ex_imp_loaddoor_02*.

Sofern noch nicht geschehen, setzen wir in die *in_impbig_blend_01* in *impsmall_doorjam_01* ein, damit die Tür auch hineinpasst ;).

Nachdem diese eingepasst wurde, klicken wir doppelt darauf, woraufhin wieder das altbekannte Fenster aufpoppt. (s. nächste Seite)

Diesmal jedoch klicken wir auf *Teleport*, wählen 'unsere' Insel *Bitterküste -3,-11* und klicken auf *Select Marker*.

Im Normalfall sollten wir nun einne schicken rosa Pfeil sehen.

Sollte das nicht so sein, betätigen wir einmal die Taste **M**, um Marker sichtbar zu machen.

Der Pfeil gibt an, wohin der Spieler teleportiert wird, wenn er die Tür aktiviert.

Damnach platzieren wir ihn vor dem Eingang unserer Burg.

Damit man auch in die Burg hineinkommt, müssen wir noch eine Teleport-Tür platzieren, diesmal mit dem Ziel *mycell*.

Damit sind wir am Ende dieses Abschnittes angelangt.

Jetzt geht es darum, die ganzen kleinen Details hinzuzufügen, die später für die Atmosphäre sorgen...

Door

ID

Script

Name

References Persist

Blocked

Reference Data

Position

X

Y

Z

3D Scale

Extra Data

Health Left

Soul

Assignable Data

Owner

Global Variable/Rank

Teleport

Load Cell

Locked

Level

Key

Trap

Reference Blocked

3.4: Feinschliff

Das Interior ist so gut wie fertig, aber das Exterior braucht noch eine Menge Arbeit, so fehlen z.B. noch die ganzen Pflanzen, die Tiere, etc...

Fangen wir direkt an und beschäftigen uns mit unseren Hausexterior, in meinem Fall die Burg.



So sieht das doch schon um einiges besser aus...

Trotzdem sieht die Landschaft immernoch recht leblos aus - was fehlt sind Pflanzen.

Diese finden sich wiedererwarten im Kateireiter "Container", denn das sind sie im Grunde genommen. (Organische-)Container für die Alchemiezutaten. Organisch deswegen, weil der Inhalt 'nachwächst', d.h. nach einiger Zeit wieder auftaucht...

Nun, wie ihr euch sicher bereits gedacht habt haben Pflanzen das Präfix *flora_...*

Setzen wir also gleich ein paar in die Landschaft sowie in das Küstenahe Wasser...

Am besten seht ihr euch etwas in MW um, auf diese Weise bekommt man oft nützliche Vorschläge wie man *flora* 'richtig' einsetzen kann...



Lichter, die unsichtbar sind.

Diese sind nur im Editor durch ein Glühlampen - Symbol zu sehen.

Nun, welche Beleuchtung würde in einen Wald passen? Eigentlich ja keine denn in einem Wald gibt es selten Lichtquellen ;-). Ausnahme sind hier Photoaktive Pflanzen wie beispielsweise 'Leuchtender *Russula*' (*flora_BC_mushroom_01* - *flora_BC_mushroom_05*), die oft ein leicht bläuliches Licht absondern... Wenn man sich im Editor einmal die Bitterküste ansieht, wird man überall solche Lichter finden...

Man erkennt zwar, was es sein soll, aber es fehlt noch eine Menge, wie beispielsweise die Schießscharten, welche *ex_imp_arrowslit* heißen.

Natürlich lassen sich auch alle anderen Statics einsetzen.

Dekorieren wir also nun nach Herzenslust unser Burg exterior, wie wir es schon mit dem Interior gemacht haben.

Zu einer Burg würden z.B. Flaggen passen. Oder eine Balliste auf dem Dach...



Auf dem Bild links habe ich auf der Lichtung einige Pflanzen verteilt.

Jedoch ist sowohl dieses Bild als auch das der Burg oben bei zugeschaltetem *Globalem Licht* gemacht worden - ohne wäre es viel zu dunkel...

Was fehlt ist also eindeutig die richtige Beleuchtung.

Nun werde ich diese nur kurz anschnitten, denn in den Abschnitten 7.3.2 und 8.4.1 werde ich mich näher dazu äußern...

Also.

Neben den Gegenständen, die die Gegend beleuchten, wie beispielsweise Fackeln, gibt es auch noch eine Reihe

Nun, wenn wir im Wald schlecht Licht einsetzen können, dann vielleicht das Gegenteil, nämlich Dunkelheit.

Schließlich stehen die Bäume recht dicht, und wenn es drumherum dunkel ist, dann sieht die Lichtung auch ohne Licht heller aus.

Nun hat auch Dunkelheit in MW eine ID, nämlich *dark_xxx* wobei *xxx* für den Radius steht. Experimentiert ein wenig mit den verschiedenen ‘Schwarzlichtern’ herum, bis ihr ein befriedigendes Ergebniss habt.

Das meine seht ihr hier. Sieht doch schon recht dunkel und bedrohlich aus ;-).

Nun wollen wir die Lichtung etwas hervorheben. Dazu eignet sich sehr gut ein ‘magisches’ blaues Licht, wie beispielsweise *blue_256_pulse_512*. *Pulse* heißt, das das Licht pulsiert, d.h. stärker und schwächer wird, was den magischen Effekt noch unterstreicht...

Am besten kommt dies Ingame raus, dennoch habe ich einen Editor Shot gemacht.

Allerdings ist der Effekt hier nur zu erahnen, daher rate ich euch das ganze einmal ingame anzugucken, dort sind die Effekte immer wesentlich besser zu sehen.

Auch die Burg will beleuchtet werden. Hier eignen sich am besten Fackeln auf oder an den Mauern.

Noch ein Tipp am Rande: Unter *View* gibt es den Punkt ‘*Light Radius*’ (oder *L*), mit dem man sich den Radius der Lichter anzeigen lassen kann.

Nun, ich würde euch gerne ein Bild der fertigen Burg präsentieren, jedoch hat sich der Editor scheinbar spontan entschieden, das Plugin zu schrotten ;-).

Das lehrt uns jedoch eine sehr wichtige Sache...

SICHERHEITSKOPIEN!

Nundenn, ihr seit jetzt in der Lage Interiors sowie Exteriors zu designen...

Herzlichen Glückwunsch!

Im nächsten Kapitel werden wir uns damit beschäftigen, wie wir eigene Gegenstände erschaffen können.



Neue Objekte



Neue Objekte bieten eine breite Fülle von Möglichkeiten.
Am besten rede ich nicht lange um den heißen Brei herum sondern fange gleich an :-)

4.1: Waffen

Wir fangen deshalb mit Waffen an, weil sie nach Activators am einfachsten zu erstellen sind, und wir so gut die Item-Erstellung lernen können.

Der erste Schritt ist, zum Reiter *'Weapons'* im Objektfenster zu wechseln.

Dann klicken wir - wer hätte das gedacht? - mit der rechten Maustaste in das Fenster und wählen *'New'*, woraufhin wir mit diesem Fenster konfrontiert werden:

Die Felder *'ID'* sowie *'Name'* sind selbsterklärend.

'Type' ist, wie ihr sicher richtig vermutet habt der Waffentyp.

Es gibt insgesamt 14 verschiedene davon, die ich jedoch nicht näher bezeichne, da sie für jeden Englischkundigen zu verstehen sein sollten.

In unserem Fall machen wir eine zweihändige Axt, daher wählen wir *'AxeTwoClose'* als Waffentyp.

Wenn wir wollen, können wir der Waffe ein Script zuweisen, wodurch es jedoch seine Stapelbarkeit verliert, man sollte es also nur für Waffen wie Schwerter, Äxte, etc einsetzen. Desweiteren werde ich in diesem Tutorial **nicht** auf Scripts eingehen, das würde den Rahmen eindeutig sprengen.

Nun, weiter im Text.

'Weight' gibt das Gewicht der Waffe an, *'Value'* ihren Wert, *'Health'* die Haltbarkeit, *'Speed'* die Geschwindigkeit mit der geschlagen wird, *'Enchantment'* die Verzauberungspunkte und *'Reach'* schließlich die Reichweite.

In dem Drop-Down-Menu *'Enchanting'* kann man eine beliebige Verzauberung auswählen.

'Ignores Normal Weapon Resistance' schließlich bestimmt, ob die Resistenz gegen normale Waffen ignoriert werden soll, beispielsweise bei Gespenstern.

Beginnen wir nun mit dem eigentlichen Erstellvorgang.

Als erstes muss eine so genannte *'Art File'* ausgewählt werden. Die *Art Files* haben in MW das Format *.nif* was für *Netimmerce* steht.

Nach einem Klick auf den Button poppt ein Fenster auf, in dem wir aufgefordert werden, ein *Mesh* daher, das Waffenmodell auszuwählen.

Da wir eine Waffe erstellen, befinden sich die Models im Ordner 'w'. Wenn ihr die Files nicht auf PC kopiert habt, so müsst ihr die Editor-CD einlegen und in den betreffenden Ordner dort wechseln.

Nun, das Model das wir verwenden hat den Namen 'w_dwemer_battleaxe'.

Als nächstes müssen wir uns ein passendes Inventarbild suchen, dazu klicken wir auf den Button 'Inventory Image'.

Auch hier wählen wir den Ordner 'w', sowie die File 'tx_dwemer_battleaxe'.

Das Icon wird nun rechts neben den beiden Buttons erscheinen.

Nun wird es Zeit, den Schaden, das Gewicht, den Wert, etc einzustellen.

Beim Schaden gibt es drei 'Sorten': *Chop*, *Slash* und *Trust*.

Ersteres ist der 'Standart-Schlag' wenn man stehenbleibt und angreift, das zweite ist der Schwung wenn man sich seitwärts bewegt und 'Trust' schließlich ein Stich.

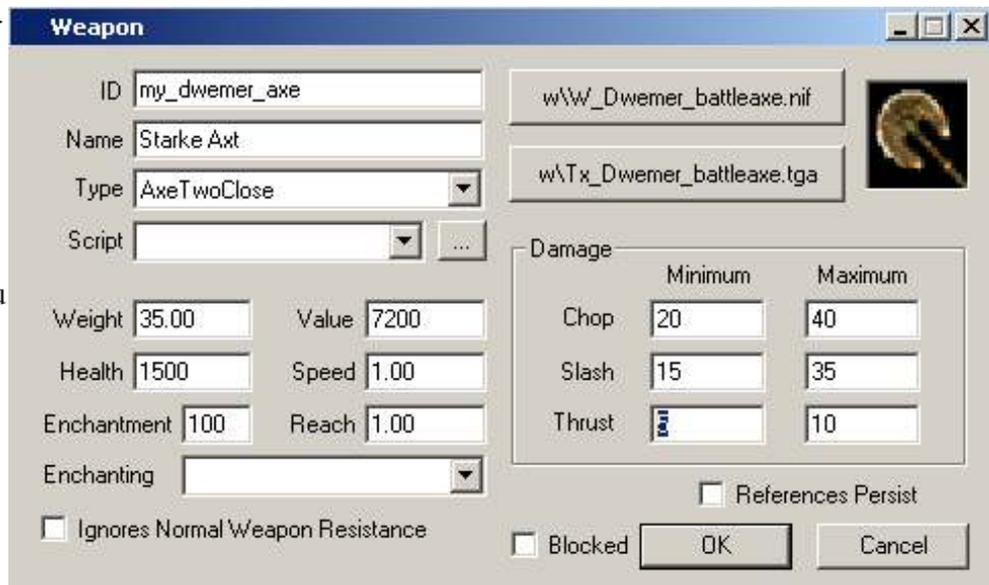
Bei der Vergabe des Schadens sollte bedacht werden, das es sich um eine Axt handelt, der größte Schaden sollte demnach bei *Chop* und *Slash* liegen.

Da es ein Beidhänder ist kann auch die Geschwindigkeit nicht alzu groß sein, umso höher jedoch das Gewicht.

Die Reichweite kann bei 1 belassen werden, der Rest ist eure Sache :-).

In meinem Fall sieht die fertige Waffe so aus:

Diese Waffe könnt ihr jetzt irgendwo in Morrowind verstecken, vielleicht in einem eigenen Dungeon, auf einer kleinen Insel? ;-)
Die richtigen Werte zu finden ist ein schwieriger Prozess, oft hilft es dabei, sich die Originalwaffen anzusehen...



4.2: Bodyparts

Okay, jetzt wird es ein kleinwenig kompliziert.

Aber wirklich nur ein bisschen. ;)

Bevor wir weiter machen und Kleidung sowie Rüstungen erstellen, ist es wichtig zu wissen, was Bodyparts sind.

Nun.

Während Creatures aus einem 'Stück' bestehen, sind NPC's aus 20 zusammengesetzt.

Den so genannten Bodyparts.

Das wären im einzelnen:

2x feet, 2x ankles, 2x knees, 2x upperlegs, 2x upperarms 2x forearm, 2x wrist, 2x hands, groin, chest, head sowie *hair*.

Auf deutsch heißt das (in der Reihenfolge von oben) soviel wie:

Füße, Knöchel, Knie, Schenkel, Oberarme, Unterarme, Schultern, Hände, Leistengegend, Oberkörper, Kopf und Haare.

Nun fragt ihr euch wahrscheinlich, warum man es nicht einfach wie bei den Creatures gemacht hat und auch NPC's aus einem Stück fertigte.

Das lässt sich am besten erklären indem ich beispielsweise erkläre, was passiert wenn man IG (InGame) einen Helm aufsetzt.

In diesem Fall wird der Helm nicht einfach 'über' den Kopf gezogen, sondern die Bodyparts *head* sowie *hair* verschwinden und werden durch den Bodypart 'Helm' ausgetauscht.

Nun gibt es aber auch offene Helme, d.h. man sieht das Gesicht.

Das wird dadurch realisiert, das nurnoch die Haare 'bedeckt' werden.

Bei anderen Objekten ist dies etwas komplizierter.

Nehmen wir das Beispiel einer Greave.

Eine Greave bedeckt 5 bodyparts, wie wir dem Objektfenster entnehmen können und zwar: *groin (g)*, *left upper leg (lul)*, *right upper leg (rul)*, *left knee (lk)* sowie *right knee (rk)*.

Wie wir jedoch sehen können werden insgesamt nur 3 Bodyparts benutzt und zwar

a_namederrüstung_greaves_k, *a_namederrüstung_greaves_ul* sowie

a_namederrüstung_greaves_g.

(**a** Steht dabei für Armor und die letzten Buchstaben für den jeweiligen Bodypart.)

Das liegt daran das alle Bodyparts, die zweimal vorkommen (*feet, ancles, knees, upperlegs, upperarms, forearms, wrist, hands*) von MW automatisch gespiegelt werden, wobei allerdings etwaigige Partikeleffekte verloren gehen.

Ein Kleidungsstück kann auch mehrere Körperteile bedecken, jedoch nur eins ersetzen.

Dies ist z.B. bei Roben der Fall. Bei einer Robe wird immer nur der Chest ersetzt jedoch Chest, left Ankle sowie right Ankle bedeckt.

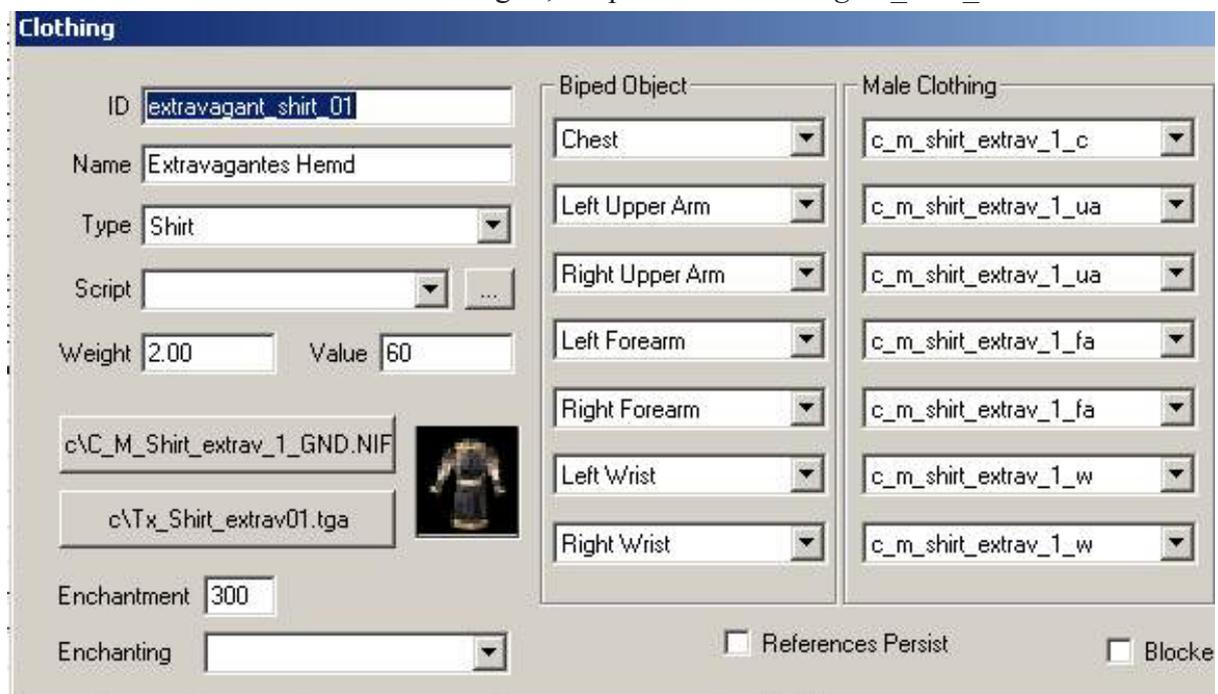
Durch spezielle Modeling-Techniken ist es möglich, das MW erkennt, welcher Teil des Meshes die Arme sind, so das die Bewegungen dieser auf das Mesh übertragen werden.

Nundenn, ich hoffe euch ist klargeworden, was Bodyparts sind und wir können mit der nächsten Stufe, Kleidung & Rüstung beginnen.

4.3: Kleidung & Rüstung

Nundenn wenden wir uns dem ersten praktischen Aufgabengebiet der Bodyparts zu: Rüstungen und Kleidung.

Sehen wir uns einmal bestehende Kleidung an, beispielsweise *extravagant_shirt_01*:



Auf den ersten Blick recht verwirrend, bei näherer Betrachtung jedoch eigentlich ganz harmlos. Neben den Dingen, die uns schon von Waffen bekannt sind auf der linken Seite, befindet sich rechts die Liste der bedeckten Bodyparts und der dafür einzusetzenden. Normalerweise steht dort nur bei Male etwas, zieht ein weiblicher NPC/Char die Kleidung an, werden diese 'männlichen' Bodyparts benutzt, sofern bei female nicht eigene angegeben sind...

Wollen wir nun beispielsweise eine neue Robe machen ist es am einfachsten eine bestehende Robe umzubennen, im darauffolgenden Fenster auf 'Yes' zu klicken und die Werte der daraufhin kreierten Kopie abzuändern.

Auch bei Armor ist dies nicht anders.

Haben wir jedoch neue Models, so müssen wir auch neue Bodyparts erstellen.

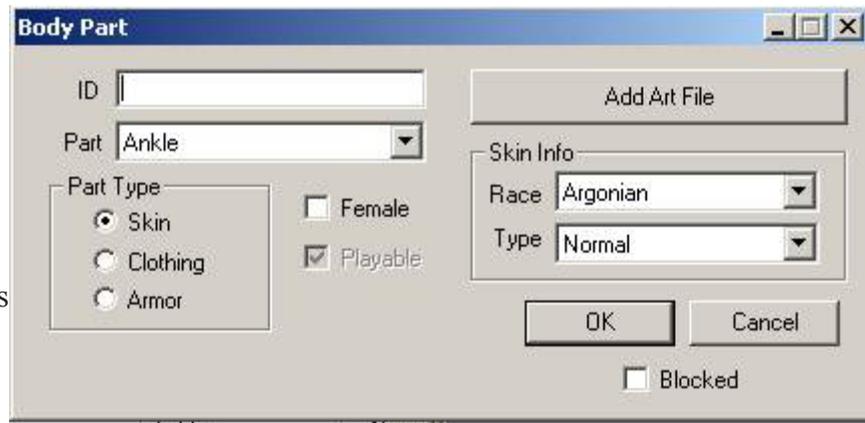
Das geht denkbar einfach.

Nachdem wir auf new geklickt haben poppt dieses Fenster auf:

ID sollte soweit klar sein, *Part* ist eben der jeweilige Bodypart und *Part Type* gibt an, was für eine Art Bodypart es ist.

Clothing sowie *Armor* sind selbsterklärend, *Skin* ist ein Teil eines Körpers.

Sollte man *Skin* wählen, so muss man rechts Rasse sowie den Typ, *Normal* oder *Vampire* (Werewolf bei Bloodmoon?) auswählen.

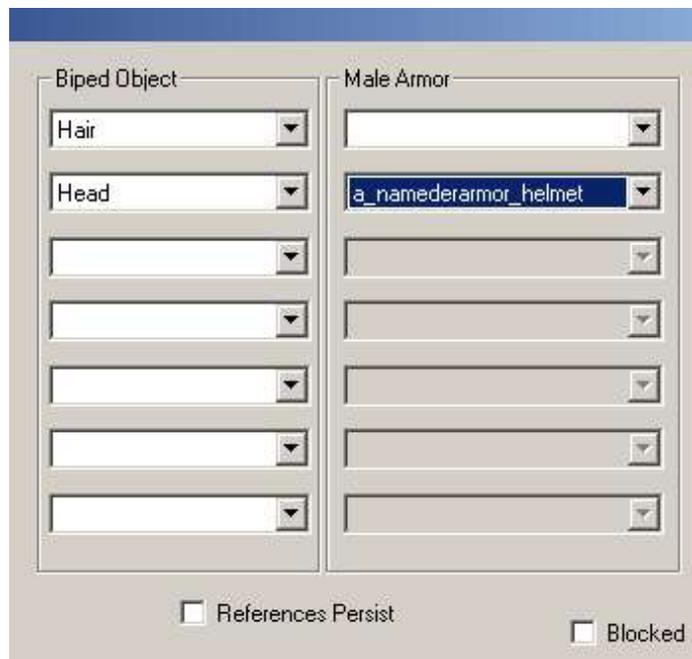


Erstellen wir dann beispielsweise einen Helm, geben wir einfach die Werte ein und wählen rechts folgendes:

...und schon haben wir einen voll funktionsfähigen Helm...

Es ist zu beachten das immer mindestens zwei Models gebraucht werden: Eins als Bodypart und eins, wenn man das Objekt ablegt...

Bei Helmen können die beiden noch identisch sein, aber haben wir es beispielsweise mit greaves zu tun, wird klar das dafür ein eigenes Objekt gebraucht wird.



4.4: Sonstiges:

Alle anderen Objekte zu erstellen, sollte eigentlich eine Kleinigkeit sein, jedoch werde ich auf einige noch kurz eingehen.

Book: Mittels *Teaches* kann man einen bestimmten Skill steigern lassen.

Der Text des Buches wird in html (!) eingegeben.

Die wichtigsten html Befehle, die für Bücher gebraucht werden werde ich im folgenden einmal kurz erläutern.

In html werden die einzelnen Befehle als sogenannte *Tag's* geschrieben. Diese Tags beginnen mit < und enden mit >.

Wer BBcode aus den WBB (WoltLab BurningBoard) Foren kennt, wird weniger Schwierigkeiten haben es zu verstehen.

```
<DIV ALIGN="CENTER"><FONT COLOR="000000" SIZE="3" FACE="Magic Cards"><BR>
```

...Steht am Anfang jedes Buches.

Das erste Tag bestimmt die Ausrichtung. Hier kann statt "CENTER" auch "LEFT" und "RIGHT" stehen.

Der zweite Tag bestimmt die Farbe der Schrift, wobei die Zahlen folgendes bedeuten:

#ffffff (weiß)

#ffff00 (gelb)

#f00000 (rot)

#000000 (schwarz)

#00ffff (türkis)

#0000ff (blau)

#00ff00 (grün)

#ff00ff (pink)

...etc

Der dritte Tag bestimmt die Größe, der vierte die Schriftart.

 schließlich ist ein Absatz.

OHNE DIESEN TAG WIRD IM BUCH KEIN ABSATZ ERSCHEINEN!

Eine weitere wichtige Funktion ist das Einfügen von Bildern.

Das wird mit folgendem bewerkstelligt:

```
<IMG SRC="Barbarian_A_AtronachStorm.TGA" WIDTH="240" HEIGHT="240">
```

Das Bild muss im Ordner *BookArt* liegen, sowie das Format *bmp*, *tga* oder *dds* haben.

Hiermit beende ich diesen kleinen Crashkurs in html, bevor ich alzuviel Scheiße verzapfe ;-).

Spellmaking/Enchanting: Die Bedienung sollte eigentlich recht intuitiv zu schaffen sein.

Mit einem Häkchen bei *AutoCalculate* werden *EnchantmentCost/SpellCost* von MW berechnet.

Der Rest sollte klar sein.

Bei *Spellmaking* kann man neben 'normalen' zaubern auch Krankheiten, Fähigkeiten etc erstellen.

Light: Hier hat man eine Menge Einstellungsmöglichkeiten.

Mit einem Häkchen bei *CanCarry* bestimmt man, ob man das Licht aufheben kann.

In diesem Fall muss ein Name, die Lebensdauer etc angegeben werden.

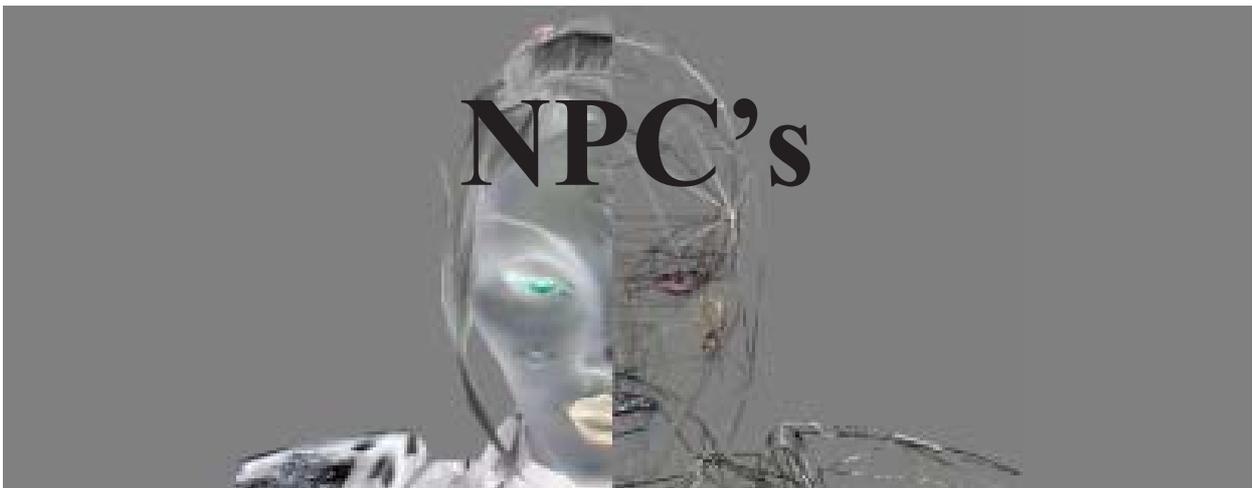
Flicker Effects sollten jedem bekannt sein, es handelt sich dabei um das heller - und dunkler werden des Lichtes, wie alle Kerzen, Fackeln etc es aufweisen.

LeveledItem: Leveled Items sind quasi Variablen. Sie können je nach Spielerlevel die Form von verschiedenen Gegenständen annehmen, die man in die Tabelle auf der rechten Seite ziehen muss.

Auch hier steht *AutoCalculate* zur Verfügung, sowie die Möglichkeit, kein Item erscheinen zu lassen.

LeveledCreature: Wie Leveled Item nur mit Kreaturen. Auf diese Weise hat man an derselben Stelle im Spiel mal einen Kwama Aufklärer und mal einen Kagouti.

Ich denke, der Rest sollte wirklich intuitiv bedienbar sein...



NPC steht für Non Player Character.

Also praktisch für die gesamte Bevölkerung von MW :-).

Sie zu erstellen und ihre Funktionen kennenzulernen ist Ziel dieses Kapitels.

5.1: Erstellen eines NPC's

Fangen wir ganz langsam an ;-).

Wenn wir im Reiter NPC im Objektfenster auf New klicken, bietet sich uns folgendes dar:

Neben den altbekannten Feldern *ID*, *Name* sowie *Script* finden wir noch ein Pull-down Menu für Rasse und Klasse sowie die Gilde samt Rang und eine Auswahl an Köpfen und Haaren...

Mit einem Häkchen bei *Essential* kann man bestimmen, ob die Meldung, das das Spiel nichtmehr erfolgreich beendet werden könne erscheinen soll, wenn der NPC getötet wurde.

Corpes Persist bewirkt, das der tote Körper ewig liegen bleibt.

Das ist wichtig wenn die Person schon von Anfang an tot ist und man ihn im laufe des Spieles findet...

Das ist z.B. beim Steuereintreiber in Seyda Neen so.

Respawn schließlich bewirkt, das der tote nach einer Weile wieder aufersteht...

Unter diesen Feldern sind die einzelnen Attribute, Fähigkeiten sowie die Farbe des Blutes einstellbar.

Will man Einstellungen daran vornehmen, so muss das standartmäßige Häkchen bei *Auto Calculate Stats* entfernt werden.

Rechts befindet sich das Inventar sowie das 'Zauberbuch' des NPC's. Hier können dem NPC per drag'n'drop Gegenstände sowie Zauber mitgegeben werden.

(Hinweis: Ob ein NPC eine Rüstung anzieht, oder eine Waffe benutzt, die man ihm ins Inventar legt, ist abhängig von seinen Fertigkeiten. Wenn er die Auswahl zwischen einem Schwert und einem Dolch hat wird er die Waffe nehmen, in dem sein Skill höher ist. Will man verhindern, das er die Rüstung anlegt, so sollte man ihm entweder eine bessere geben oder man wählt eine zivile Klasse wie z.B. *Trader*.)

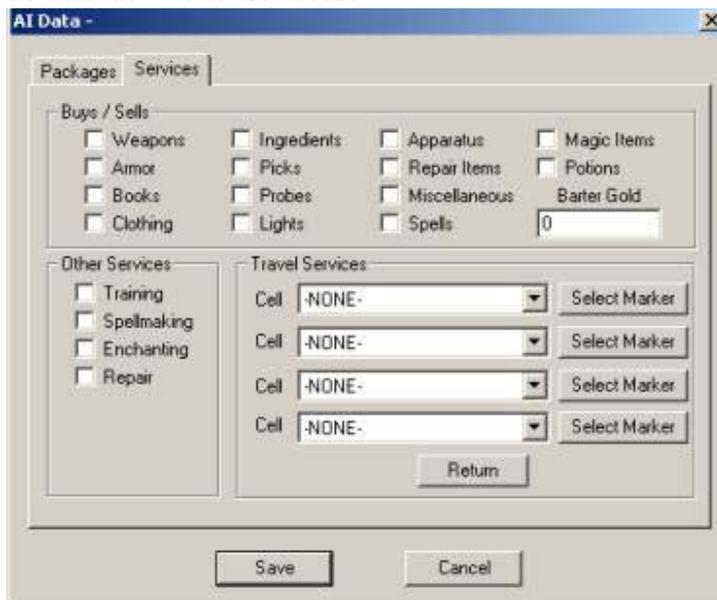
Eigentlich sollte die Erstellung eines NPC's nun kein Problem sein.
Einfach Rasse Klasse Fraktion und Level auswählen (oder die Skills manuell verändern, wenn man möchte), fertig.
Der nächste Schritt, ein NPC der seine Dienste anbietet, kommt jetzt ;-)

5.2: Service NPC's

Unter 'Service NPC's' verstehe ich NPC's, die verschiedenste Dienste anbieten, z.B. Reisen oder Reparieren....

Eigentlich sind diese NPC's ganz simpel zu erstellen...

Nachdem man das Häkchen von *Auto Calculate Stats* entfernt hat, klickt man auf den Button *AI* und dort auf den Reiter *Services*.



...und schon haben wir alles was wir brauchen.

Bei *Buys/Sells* kann man einfach einstellen mit welchen Gegenständen der Händler handeln soll.

Barter Gold ist das Geld, das der Händler von vornherein zu Verfügung hat.

Other Services ist eigentlich selbsterklärend.

Mit *Training* trainiert der NPC seine drei besten Skills, *Spellmaking* sollte klar sein, ebenso wie *Enchanting* und *Repair*.

Mittels *Travel Services* kann man vier Orte einstellen, zu denen einen der NPC bringen kann.

Wie bei den Teleporttüren einfach den

Zielort im Pull-Down-Menu auswählen und auf *Select Marker* klicken - zurück zum NPC geht mit einem Klick auf *Return*.

5.3: AI

AI steht für ‘*Artificial Intelligence*’, auf deutsch ‘*Künstliche Intelligenz*’ also *KI*.

Die AI steuert das Verhalten der Lebewesen in MW.

Sie sagt, wann Kreaturen sich bewegen und wann sie angreifen sollen.

Sie steuert die NPC’s auf ihren Bahnen entlang durch die Stadt, etc pp.

Im Fall der NPC’s gibt es insgesamt 5 verschiedene ‘*AI-Packages*’: *Travel*, *Follow*, *Wander*, *Activate* und *Escort*.

Ich werde jedes dieser ‘*Pakete*’ im folgenden näher erläutern und zeigen, wie man sie richtig benutzt.

Man findet sie, wenn man - wer hätte es gedacht - im NPC Bildschirm auf AI klickt.

Die Packages werden von oben nach unten abgearbeitet. Sollte der NPC dabei gestört werden, z.B. durch einen Angriff, hört er so lange damit auf, die Packages abzuarbeiten, bis der Angreifer geflohen oder Tod ist.

5.3.1: Wander

Distance:	128	Duration: (in hours)	5
		Time of Day: (?:00)	0
Idle Chances			
Idle 2:	60	Idle 6:	0
Idle 3:	20	Idle 7:	0
Idle 4:	10	Idle 8:	0
Idle 5:	10	Idle 9:	0
Save		Cancel	

Ich beginne mit *Wander*, da dies eins der ‘*leichteren*’ AI-Pakete ist und uns daher einen guten Einstieg erlaubt.

Wander sorgt dafür, das der NPC auf einer zufälligen Route in einem bestimmten Radius herumläuft.

Distance bestimmt eben diesen Radius, *Duration* die Dauer die der NPC herumläuft und *Time of Day* schließlich bestimmt, wann der NPC ‘*loslegt*’ und beginnt herumzulaufen. *Idle Chances* bestimmt, mit welcher Wahrscheinlichkeit eine bestimmte ‘*Idle-Animation*’ ausgeführt wird. Am besten ist es wie immer, wenn ihr ein wenig mit den

Einstellungen herumexperimentiert und guckt was passiert ;-).

(**Hinweis:** Neue NPC’s haben grundsätzlich ein ‘*Wander-Package*’. Will man also NPC’s haben, die stillstehen, wie beispielsweise Torwächter, so muss das Paket bei ihnen gelöscht werden.)

5.3.2: Travel

Travel ist ein enorm praktisches Paket. Mit ihm kann man einen NPC von a nach b laufen lassen.

Am besten lässt sich das durch ein Beispiel verdeutlichen.

Wir erstellen ganz normal einen NPC und setzen ihn z.B. vor die Brücken zu Balmora.

Entfernen wir nun als erstes das *Wander-Package* und klicken dafür bei ‘*Add Package*’ auf *Travel*.

Es erscheint nun ein Fenster mit Koordinaten, die wir so übernehmen.

Als nächstes doppelklicken wir auf *Travel* und dann auf ‘*View Location*’.

Die Kamera springt nun zu einem großen roten X. Dieses X ist die Stelle, zu der der NPC sich bewegen wird und lässt sich wie gewohnt vertikal und horizontal bewegen.

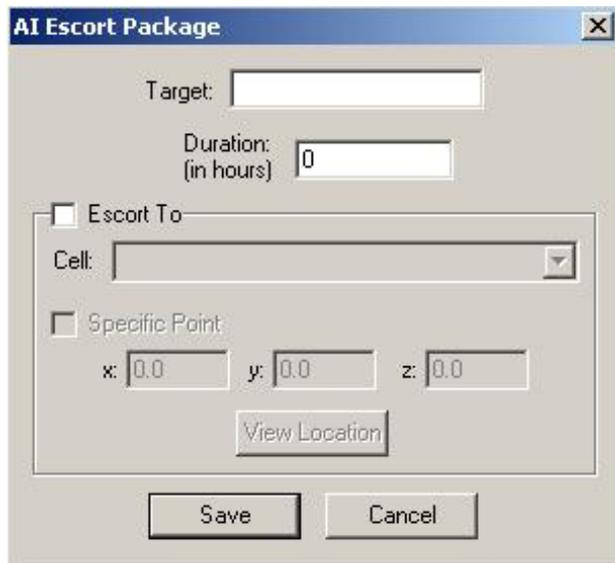
Erstellen wir nun weitere *Travel-Pakete* und lassen den NPC so nach Balmora gehen.

Bei Tests wird euch auffallen, das das sogenannte ‘*Pathfinding*’ der NPC’s nicht alzu gut ist, das heißt, das sie des öfteren einmal steckenbleiben oder von einer Brücke fallen, wenn man nicht aufpasst.

Sobald alle *Travel-Pakete* abgearbeitet wurden, bleibt der NPC stehen.

5.3.3: Escort

Mit dem *Escort-Package* veranlasst man einen NPC dazu, eine Person zu schützen. Nachdem wir es erstellt haben, sehen wir folgendes Fenster:



Target: Bestimmt das Ziel, das eskortiert werden soll. Dieses sollte auf jedenfall einzigartig sein, da sonst **alle** Referenzen eskortiert werden.

Duration: Wie lang der NPC das Ziel eskortieren soll (gilt nicht wenn 'Escort To' gewählt wurde).

Escort To: Wohin das Ziel eskortiert werden soll

Specific Point: Sofern *Escort To* gewählt wurde kann man hier ähnlich wie beim *Travel-Paket* einen genauen Punkt wählen.

(**Hinweis**: Sollte ein anderes Ziel als der Spieler (ID: Player) gewählt werden, muss zusätzlich ein *Follow-Paket* eingerichtet werden, da der NPC sonst einfach stehen bleibt.)

5.3.4: Follow

Können wir recht schnell abhandeln, da es praktisch genauso ist wie bei *Escort*.

Ehrlich gesagt weiß ich nicht warum es sowohl *Escort* als auch *Follow* gibt, da sie eigentlich fast das gleiche bewirken...

5.3.5: Activate

Ein recht mächtiger Befehl.

Mit ihm kann man den NPC veranlassen, Dinge zu aktivieren. Das einzige was angegeben werden muss, ist die ID des Ziels. Im einzelnen wird der NPC folgendes tun:

- | | |
|--------------------------------------|--------------|
| Tür/Container | - öffnen |
| NPC's | - ansprechen |
| Gegenstände die auf dem Boden liegen | - aufnehmen |
| Bücher/Schriftrollen | - lesen |

Also im Prinzip dasselbe, was auch der Char tut, wenn man auf irgendetwas klickt.

5.4: Animation

Neben den Buttons *AI* und *Dialogue* gibt es noch einen dritten, *Animation*.

Wofür dieser Button gut sein kann sehen wir in 'Dreseles Haus der irdischen Freuden' in Suran, wo einige halbnackte NPC's tanzen.

Klicken wir bei ihnen einmal auf *Animation*, so sehen wir, das alle Animatione bis auf *Idle9* gelöscht wurden, sodass sie praktisch nichts anderes mehr können, als zu tanzen.

Nun wollen einige in ihrem Haus auch solche tanzenden Frauen (oder Männer) haben.

Aber es funktioniert nicht, da sie eine Kleinigkeit übersehen...

Das *Wander-Paket*.

Wir erinnern uns, das dort neben dem Radius etc auch die *Idle Chances* angegeben waren.

Standartmäßig ist die für *Idle9* 0.

Sie *können* garnicht tanzen, weil die wahrscheinlichkeit dazu 0 beträgt ;-).

Dialoge

Dialoge sind ein wichtiger Bestandteil vieler Mod's.

Auch ist mit ihnen mehr möglich, als es auf den ersten Blick scheint. So kann man einen NPC durch Dialoge beispielsweise dazu veranlassen, den Spieler anzugreifen. Oder man entfernt Gegenstände aus dem Inventar. Oder man macht Tagebuch-Einträge. Oder...

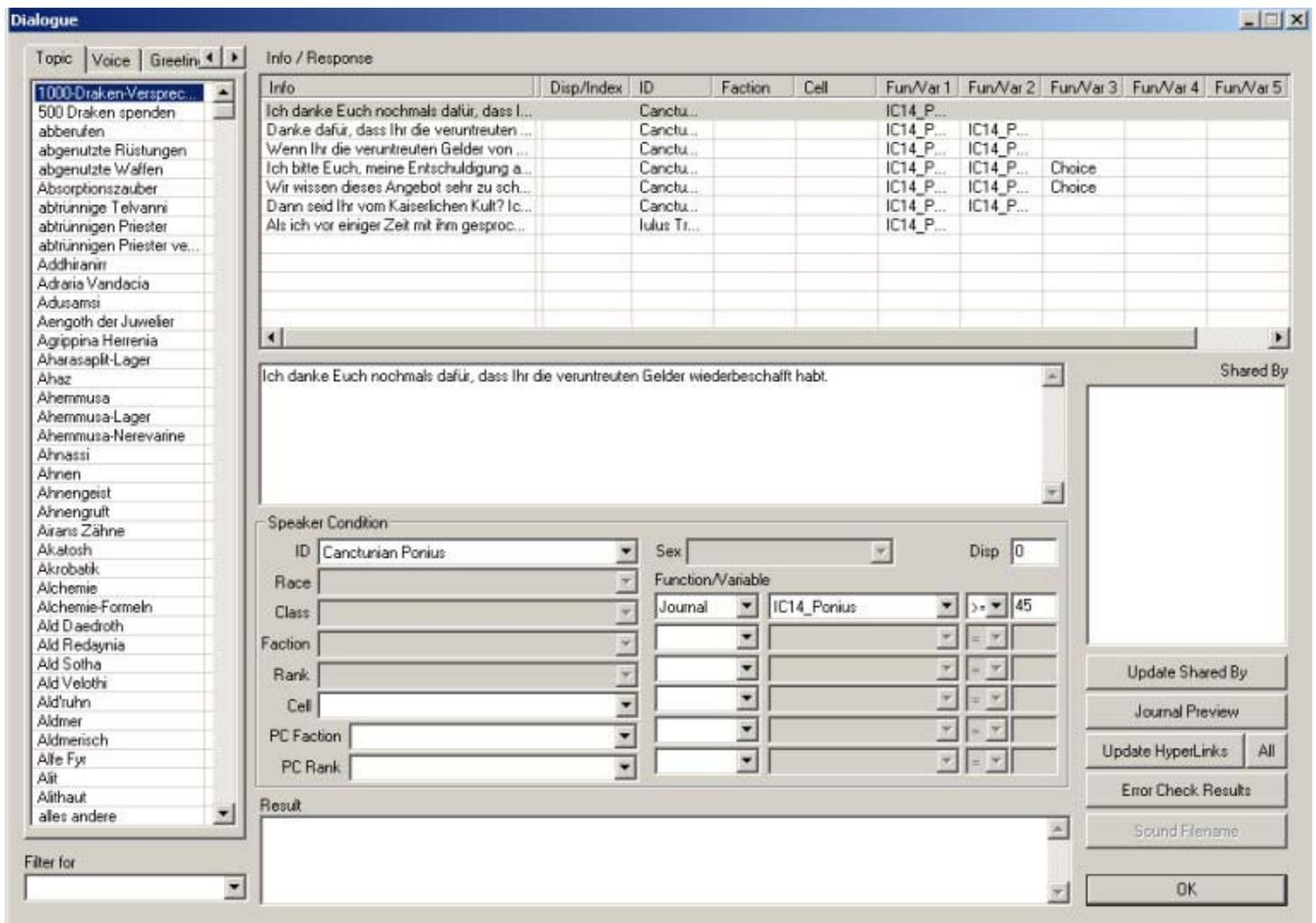
6.1: Das Dialogue - Window

Es gibt zwei Möglichkeiten, zum Dialogue-Window zu kommen.

Die erste ist, in einem NPC-Fenster auf den gleichnamigen Button zu klicken, wodurch gleichzeitig ein Filter aktiviert wird.

Die zweite ist über Character->Dialogue.

Wie auch immer wir es gemacht haben, wir finden nun folgendes Fenster vor:



Auf der linken Seite befindet sich die Liste der sogenannten *Topics*, das sind die einzelnen Themen die man im Spiel ansprechen kann. Durch die Karteireiter an der oberen Seite ist es möglich, zwischen den verschiedenen Bereichen *Topic*, *Persuasion*, *Greeting*, *Voice* sowie *Journal* umzuschalten.

An der untereren Seite kann man einen *Filter* einstellen, der bewirkt, das alle Themen, die nichts mit dem NPC, dessen ID man auswählt zu tun haben, ausgeblendet werden.

Auf der rechten Seite schließlich befindet sich unser Arbeitsplatz.

Hier können wir die verschiedenen Antworten auf das Thema einstellen, je nach dem ob die *Speaker Conditions* die sich unten befinden eingehalten werden.

Ganz unten schließlich ist das *Dialogue Result Window*, mit dem wir uns später noch genau befassen werden.

Am besten fangen wir gleich an, anstatt das ich mir an Beschreibungen die Zähne ausbeiße, wo man es doch viel besser mittels *Learning-by-doing* versteht.

6.2: Topics

Ersteinmal sollte geklärt werden, was ein Topic eigentlich ist.

Es handelt sich dabei um eine Art *Kontainer*, in dem die verschiedenen Antworten enthalten sind.

Welche Antwort kommt, hängt von den eingestellten *Speaker Conditions* ab.

Ein Beispiel: Wir wollen eine Quest machen.

Das erste Mal, wenn der Spieler das Topic anklickt, soll er also gefragt werden, ob er den Auftrag annimmt, was mit *Multiple Choice* möglich ist.

Daraufhin bekommt er ein Tagebucheintrag.

Hat er den Job erledigt, bekommt er einen weiteren.

Klickt er das nächste Mal auf das Topic, wird geprüft, ob der neue Tagebucheintrag vorhanden ist.

Wenn dem so ist so wird eine betreffende Antwort ausgegeben.

Man hätte das auch anders als durch einen Tagebucheintrag lösen können, beispielsweise durch Abfrage, ob ein Gegenstand vorhanden ist, etc...

6.2.1: Erstellen von Topics

Erstmal müssen wir wissen, wie wir ein Topic eigentlich erstellen.

Das geht eigentlich wie alle andere Dinge auch Rechtsklick->New. Ich nenne es in meinem Fall test.

Es sollte allerdings bedacht werden, das der Topicname der Name ist, der im Spiel auftaucht, also sollte er Sinn machen und gleichzeitig möglich einzigartig sein, um Komplikationen mit anderen Mod's vorzubeugen.

Nun haben wir also ein Topic. Aber dieses hat keinen Inhalt wie man im *Response-Window* sehen kann.

Erstellen wir auch dort ein neues, der Name ist egal, da er sich sowieso ändert, wenn man den eigentlichen Inhalt tippt, was wir auch gleich tun.

In meinem Fall ist das "*Dies ist ein Test*".

Wie bringen wir dieses Topic nun ins Spiel?

Eine Möglichkeit wäre es, in irgendeinem bereits bestehenden Gespräch das Wort irgendwie einzubinden, sodass ein sog. *Link* entsteht. (Das sind in MW die Blau unterlegten Wörter)

Die zweite wäre schließlich durch ein Script das wie folgt lauten muss:

```
Begin my_test_topic_script  
  
    player->addtopic Test
```

```
End
```

Dieses Script wird gespeichert und auf den NPC gelegt...

Die dritte Möglichkeit schließlich ist durch ein Greeting, welches ich in Abschnitt 6.3 näher beschreiben werde.

Bis dahin nehmen wir behelfsmäßig eine der oberen Lösungen.

6.2.2: Conditions:

Nun wollen wir aber meist nicht, das es auf ein Topic nur eine Antwort gibt.

Andere NPC's oder Rassen haben vielleicht eine andere Meinung über das Thema.

An dieser Stelle benötigen wir zwingend *Conditions*, um dem Spiel zu sagen, wann welche Antwort fällig ist.

Nehmen wir an, wir wollten unser Topic so abändern, das je nach Beliebtheit andere Antworten gegeben werden.

Dazu erstellen wir 2 weitere Antworten, eine positive und eine negative. Bei mir wären das.

“*Das geht dich nichts an, Drecksmaße!*”

“*Oh, wir spielen nur etwas mir Conditions herum.*”

Um festzulegen, wann welche Antwort kommt, müssen wir bei *Speaker Conditions* neben *Disp* jeweils einen Wert eintragen, beispielsweise 25, 50 und 75.

Nun wird je nach Beliebtheit eine andere Antwort gewählt.

Um das ganze noch vielfältiger zu machen, erstellen wir eine Antwort für Argonier und eine für Kahjit, die der Grammatik nunmal nicht wirklich mächtig sind.

Also zwei weitere Antworten erstellen, z.B:

“*Ein Test dies ist.*”

“**schnurr* Nur ein Test.*”

Als Condition wählen wir bei *Race* einmal Argonian und einmal Kahjit. Man kann außerdem wählen:

<i>ID</i>	- Die ID des Sprechers.
<i>Class</i>	- Die Klasse des Sprechers... Krieger antworten u.U. etwas anderes als Magier.
<i>Faction</i>	- Die Fraktion, die der Sprecher angehört...
<i>Rank</i>	- Den Rang, den der Sprecher bekleidet.
<i>Cell</i>	- Die Zelle, in der sich der Sprecher befindet.
<i>PC Faction</i>	- Die Fraktion des Spielers... evtl ist der Sprecher einer Fraktion feindlich gesinnt.
<i>PC Rank</i>	- Der Rang des Spielers - ein Gelehrter wird anders behandelt als ein wahrer Meister.

Ich denke ihr habt nun verstanden, wie die Conditions auf der linken Seite funktionieren, kommen wir nun kurz zu denen auf der rechten. Mit ihnen lassen sich folgende Dinge prüfen:

<i>Function</i>	- Hiermit lassen sich Dinge wie Bekanntheit oder die Fähigkeiten des Spielers abfragen.
<i>Global</i>	- Globale Variablen
<i>Local</i>	- Lokale Variablen
<i>Journal</i>	- Hiermit lässt sich prüfen, ob bestimmte Tagebuch-Einträge vorhanden sind.
<i>Item</i>	- Prüft, ob der Spieler einen Gegenstand im Inventar führt.
<i>Dead</i>	- Prüft, ob bestimmte NPC's Tod sind.
<i>NotID</i>	- Macht die Aussage nur, wenn der Sprecher nicht die angegebene ID besitzt
<i>NotFaction</i>	- Dasselbe wie <i>NotID</i> , allerdings mit Fraktionen
<i>NotClass</i>	- mit Klassen
<i>NotRace</i>	- mit Rassen
<i>NotCell</i>	- mit Zellen
<i>NotLocal</i>	- und lokalen Variablen

Dahinter kann man dann aus einer Reihe von *Operatoren* auswählen. Das wären im einzelnen:

=	- Der Wert muss genau <i>x</i> sein.
!=	- Der Wert muss ungleich <i>x</i> sein.
>	- Der Wert muss größer als <i>x</i> sein.
>=	- Der Wert muss größer als oder gleich <i>x</i> sein.
<	- Der Wert muss kleiner als <i>x</i> sein.
<=	- Der Wert muss kleiner als oder gleich <i>x</i> sein.

Hinter diesen Operatoren kann man schließlich den eigentlichen Wert eingeben.

Um unser Beispiel von oben fortzuführen, lassen wir prüfen ob der Spieler ein bestimmtes Objekt besitzt, in meinem Fall einen Laptop (Muss natürlich erst erstellt werden).

Ich gebe also an `Item | my_laptop | >= | 1`.

In diesem Fall benutze ich deswegen `>=`, da der Spieler aus irgendeinem Grund auch mehr als einen `Laptop` besitzen könnte.

Nun am besten experimentiert ihr auch hier ein wenig herum ;-).

6.2.3: Multiple Choice

Manchmal hat man in Dialogen Wahlmöglichkeiten, die die Reaktion des Gegenübers beeinflussen. Oft gibt es Schwierigkeiten, diese zu erstellen, also werde ich hier kurz darauf eingehen.

Als erstes braucht man etwas, auf das der Spieler reagieren soll.

Nehmen wir hier unsere Test-Topic.

Um eine Multiple Choice zu erstellen muss bei allen Antworten in das *Dialogue-Result-Fenster* folgendes eingetragen werden:

```
Choice "Reaktion 1." 1 "Reaktion 2." 2
```

Also in unserem Beispiel etwa so:

```
Choice "Tests sind immer nützlich." 1 "Ich HASSE Tests." 2
```

Als letztes müssen wir nur noch für jede Wahl einen Response Text erstellen. Als Condition wählen wir: `function | Choice | = | x`

WICHTIG: Die Texte müssen mit den Cursortasten **über** den Response-Text, in dem die Frage steht, befördert werden, da immer der erste Response Text, der alle Bedingungen erfüllt gewählt wird.

6.2.4: Dialogue Result

Im oberen Kapitel haben wir es bereits einmal verwendet, aber es kann mehr als simples Multiple Choice. Prinzipiell können alle Scriptbefehle, die nur über eine Zeile gehen, genutzt werden. Ich habe die wichtigsten, ihre Bedeutung sowie ihre Syntax im folgenden einmal aufgelistet.

AI

<code>AiActivate</code> , [ObjectID]	- Siehe Seite 31: Activate
<code>AiEscort</code> , [ActorID], duration, x, y, z	- Siehe Seite 31: Escort
<code>AiEscortCell</code> , [ActorID], [CellID], duration, x, y, z	- Siehe Seite 31: Escort
<code>AiFollow</code> , [ActorID], duration, x, y, z	- Siehe Seite 31: Follow
<code>AiFollowCell</code> , [ActorID], [CellID], duration, x, y, z	- Siehe Seite 31: Follow
<code>AiTravel</code> , x, y, z	- Siehe Seite 30: Travel
<code>AiWander</code> , range, duration, time [Idle1], ..., [idle9]	- Siehe Seite 30: Wander

Combat

<code>StartCombat</code> , ID	- Greift einen NPC oder den Spieler an
<code>StopCombat</code>	- Beendet den Angriff

Dialogue

<code>AddTopic</code> , [TopicID]	- Fügt dem Spieler ein Topic hinzu
<code>Goodbye</code>	- Beendet den Dialog (man kann nun noch Bye wählen)
<code>Journal</code> , ID, Index	- Fügt einen Tagebuch-Eintrag hinzu
<code>Choice</code>	- Siehe Seite 35: Multiple Choice

Faction

Lower Rank	- Senkt den Rank des NPC's
RaiseRank	- Erhöht den Rank des NPC's
PCExpell, [FactionID]	- schließt den Spieler aus der Fraktion aus
PCJoinFaction, [FactionID]	- Lässt den Spieler sich einer Fraktion anschließen
PCLowerRank	- Senkt den Rank des Spielers
PC RaiseRank	- Hebt den Rank des Spielers

Item/Object

AddItem, [ObjectID], count	- Fügt der jeweiligen Person x Objekte hinzu
RemoveItem, [ObjectID], count	- Entfernt x Objekte von der Person
Equip, [ObjectID]	- Lässt die Person das Objekt anlegen

Magic

AddSoulGem, [CreatureID], [SoulgemID]	- Einen Seelenstein hinzufügen
AddSpell, [SpellID]	- Einen Zauber hinzufügen
AddSpelleffects, [SpellID]	- Zaubereffekte eines Zaubers hinzufügen
RemoveSoulGem, [CreatureID]	- Einen Seelenstein entfernen
RemoveSpell, [SpellID]	- Einen Zauber entfernen
RemoveSpellEffects, [SpellID]	- Zaubereffekte eines Zaubers entfernen.

Misc

GetPCCell, [CellID]	- Den Spieler teleportieren
gotoJail	- Den Spieler ins Gefängnis stecken

Stats

Set[Stat],value	- Ein Attribut oder eine Fähigkeit auf einen bestimmten Wert setzen
Mod[Stat], mod	- Punkte zu einem Attribut oder einer Fähigkeit hinzufügen oder entfernen

Viele der o.g. Funktionen benötigen ein [ActorID]-> vor dem eigentlichen Befehl, beispielsweise player->AddSpell, [SpellID].

Ein Beispiel für ein Dialogue Result könnte so aussehen (Kommentare durch // gekennzeichnet):

```
Goodbye // Der Dialog wird beendet
StartCombat, Player // Der Spieler angegriffen
Journal, my_playercombat_entry, 1 // Ein Tagebucheintrag hinzugefügt
```

6.2.4: Text Defines

Text Defines sind Variablen, die in Dialogen zum Einsatz kommen.

Ich habe sie hier einmal aufgelistet und kurz erklärt:

%Name	- Der Name des NPC's
%PCName	- Der Spielername
%Race	- Die Rasse des NPC's
%PCRace	- Die Spielerrasse
%Class	- Klasse des NPC's
%PCClass	- Spielerklasse
%Faction	- Fraktion des NPC's
%Rank	- Rank des NPC's
%PCRank	- Rank des Spielers in der Fraktion des NPC's
%NextPCRank	- Nächster Rank des Spielers in der Fraktion des NPC's
%Cell	- Aktuelle Zelle

6.3: Greetings

Greetings (Begrüßungen) spielen eine relativ große Rolle.

Zum einen ist es damit möglich, eine Art 'Filter' zu bauen, sodass bestimmten Rassen, Klassen, etc von vornherein das Gespräch verweigert wird.

Zum anderen kann man damit Topics 'sauber' in das Spiel einfügen.

Da ihr bereits einen Großteil über Dialoge, Conditions, etc wisst, kann ich diesen Abschnitt kurzhalten.

Nachdem das Dialogue Fenster offen ist, klicken wir auf den Kateireiter 'Greetings'.

Dort sehen wir 10 'Topics', Greeting 0 - Greeting 9. Das ist nötig, um die Ordnung aufrecht zu erhalten. Jeder dieser 'Ordner' hat seinen spezifischen Inhalt:

- Greeting 0 - Alles was mit Verbrechen zu tun hat.
- Greeting 1 - Personen, Fraktionen oder Cell-bezogene Begrüßungen
- Greeting 2 - Alles was mit Vampiren zu tun hat
- Greeting 3 - Hier steht interessanterweise nur eine Begrüßung zum Thema Morag Tong
- Greeting 4 - Krankheiten, Pest sowie die Morag-Tong-Auftrag-bezogenen Begrüßungen
- Greeting 5 - Größtenteils Questbezogene Begrüßungen
- Greeting 6 - ?
- Greeting 7 - Die normalen Begrüßungen...
- Greeting 8 - Begrüßungen mit Anspielung auf die Ausrüstung des Spielers
- Greeting 9 - Rein Cell-bezogene Begrüßungen

Nun, der Rest ist wie immer. Ihr könntet eure Begrüßung in jedes dieser Topics hineintun, der Übersicht halber solltet ihr sie jedoch nach o.g. Kriterien ordnen.

6.4: Voice

Vielleicht macht ihr eine etwas aufwendigere Mod, und ihr wollt eigene Voice-Greetings erstellen.

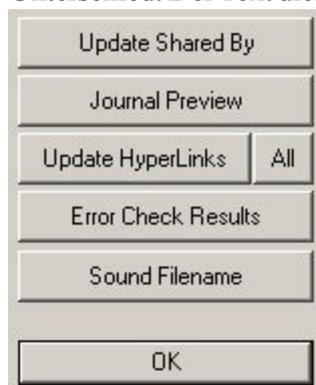
Oder das ewig gleiche 'Sterbt, N'wah!' kotzt euch an?

Dann müsst ihr mit Voice arbeiten! :-)

Auch dieses hat einige 'Untertopics', die da wären:

- Alarm - Ist interessanterweise leer
- Attack - Hier befinden sich so nette Dinge wie 'Sterbt!' 'Ihr N'Wah!', etc
- Flee - Hier befindet sich das Winseln um Gnade ;-)
- Hello - Wenn ihr in der Nähe eines NPC's seid, werden diese Begrüßungen abgespielt
- Hit - Wenn man einen NPC trifft
- Idle - Dinge, die die NPC's so vor sich hinsprechen wenn sie glauben alleine zu sein
- Intruder - Wenn man von Feinden entdeckt wurde, werden diese Dinge abgespielt
- Thief - Wenn man etwas geklaut hat und erwischt wurde

Das erstellen von Voice läuft eigentlich genauso ab wie bei normalen Responses, mit einem kleinen Unterschied: Der Text dient wirklich nur zur Beschreibung.



Hat man nun seine Soundfile, so steckt man diese als *.mp3 Datei in den Ordner *Data Files/Sound/Vo* und klickt dann im Dialogue Fenster auf den Button 'Sound Filename'.

6.5: Journal

Journalentrys (Tagebucheinträge) sind sehr wichtig in MW. Sie dienen nicht nur dem Spieler als Hilfestellung sondern auch dem Programm, zu erkennen, welche Quests bereits auf welche Weise gelöst wurden. Auf diese Weise wird z.B. erkannt, ob Dagoth Ur bezwungen wurde.

Journals funktionieren ein klein wenig anders als 'normale' Topics.

Auch hier gibt es sowas wie 'Topics', d.h. Ordner, in denen die spezifischen Journal-Einträge gespeichert sind. Die Unterscheidung zwischen den einzelnen Einträgen jedoch findet nicht durch die Conditions statt sondern durch einen sogenannten *Index*.

Dieser wird direkt per Scriptbefehl oder Dialogue Result abgerufen, z.B.: Journal My_test_journalentry, 10.

Diese Einträge können dann als Condition benutzt werden...

6.6: Persuasion

Hier stehen die Reaktionen der NPC's auf bestechen, schmeicheln, spotten oder drohen sowie Info und Service Refusal.

Ich werde im folgenden die einzelnen Topics kurz beschrieben, jedoch nicht näher darauf eingehen, das habe ich bereits zu genüge getan.

AdmireFail	- Fehlgeschlagene Bewunderung
AdmireSuccess	- Erfolgreiche Bewunderung
BribeFail	- Fehlgeschlagene Bestechung
BribeSuccess	- Erfolgreiche Bestechung
Info Refusal	- Verweigern von Informationen
IntimidateFail	- Fehlgeschlagene Einschüchterung
IntimidateSuccess	- Erfolgreiche Einschüchterung
Service Refusal	- Verweigern von Diensten
TauntFail	- Fehlgeschlagenes Spotten
TauntSucess	- Erfolgreiches Spotten

Exteriordesign für Fortgeschrittene

In Kapitel 3 habe ich bereits eine kurze Einführung in das Landscaping gegeben. Diese bechränkte sich jedoch ausschließlich auf die Grundlagen, die nötig sind, z.B. eine eigene Insel zu formen.

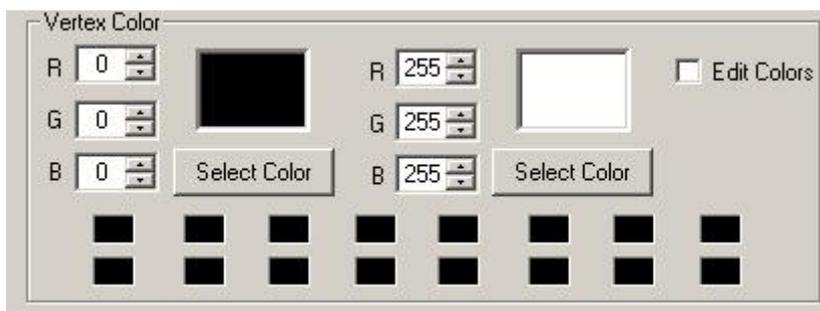
In diesem und dem folgenden Kapitel werde ich viele der Tricks und Kniffe, auf die ich während meiner Mod(der) Tätigkeit stieß, mit euch teilen. Dabei ist zu erwähnen, das ich bestenfalls besserer Durchschnitt bin. Jedoch bin ich recht aktiv in mehreren Foren und sammle dort fleißig neue Ideen, die ich hier hinzufügen kann.

Betreffende Personen werden selbstverständlich erwähnt.

7.1: Möglichkeiten von Vertex Paint

Ich habe Vertex Paint bis jetzt nur am Rande erwähnt.

Öffnen wir das Landscaping Fenster, sehen wir am unteren Rand folgendes:



Im Grunde genommen ist es nichts anderes wie in jedem anderen Malprogramm auch. Man hat zwei Farben - eine auf der linken und eine auf der rechten Maustaste, wobei weiß eine Art 'Radiergummi' ist. Die Größe wird durch den Edit

Radius bestimmt. Aktiviert wird das ganze durch einen Haken bei 'Edit Colors'.

Bleibt nur noch die Frage, wofür das ganze gut ist...

Nun, eine Möglichkeit ist, damit Schatten zu malen, beispielsweise unter Mauern etc, da Statics von Natur aus keinen Schatten werfen.

Das sieht man im Editor auch sehr gut z.B. an den Steinen im Wasser von Bittercoast...

Hier noch eine Impression und ein paar Vorschläge was man noch damit anstellen könnte:



Wie wärs z.B. irgendetwas Mystisches Wesen zu verfolgen indem man seiner Blutspur folgt?

Oder wie hier ein paar aufgepießte Körper und eine Blutspur zu einer Höhle?

Ich bin mir sicher es gibt auch noch 'friedlichere' Möglichkeiten Vertex Paint einzusetzen die mir jetzt nur nicht einfallen *g*.

Wie ihr seht ist eurer Fantasie einmal mehr kaum eine Grenze gesetzt. ;-).

7.2: Wälder

Wälder sind eine heikle Sache.

Zum einen fressen sie - besonders im Editor - viele Ressourcen. Zum anderen ist es sehr schwer, einen Wald vernünftig hinzubekommen.

Um die Anfangsschwierigkeiten, die man hat zu minimieren, werde ich euch hier einige Tipps geben...

7.2.1: Grundlegendes

Ersteinmal ist zu unterscheiden, *was* für einen Wald man bauen will.

Es macht einen großen Unterschied ob man einen sterbenden, kranken Wald, einen Mangrovenwald oder einen saftigen grünen Wald baut.

Je nachdem was man nun erreichen will, kann man die Objekte auswählen, die benutzt werden können.

Das sind allerdings nicht nur Bäume.

Auch Pflanzen oder Creatures gehören natürlich dazu.

Nundenn, haben wir uns nun entschieden geht es erstmal daran ein geeignetes Landscape anzufertigen. Wenn ihr Ideen dazu braucht, solltet ihr in Erwägung ziehen, einmal aufzustehen und den nächsten Wald aufzusuchen ;-).

Behaltet dabei im Kopf, das das Landscape Auswirkungen auf den Bewuchs hat. Auf steinigem Untergrund werden weniger aber dafür robustere Pflanzen wachsen, während in Sumpfgenden eher Mangrovenwälder entstehen dürften...

Nun glücklicherweise ist *Morrowind* aber ein Fantasy-Spiel und man hat dementsprechend viele Freiheiten. Ein Wald könnte beispielsweise trotz anhaltender Trockenheit mit Magie am Leben erhalten werden... Dann sollte man jedoch nicht auf ein bläuliches Schimmern verzichten ;-).

Steht also das Landscape, gehts an die Bäume. Beachtet hierbei das zu viele Bäume Performance-Schwierigkeiten verursachen können. Es gibt mehrere Möglichkeiten dem entgegenzuwirken. Zum einen kann man sich natürlich neue Bäume, die weniger Poly's haben aus dem Internet herunterladen.

Auch kann man die Bäume vergrößern, sodass man sie weiter auseinanderstellen kann...

Nundenn, beim aufstellen der Bäume solltet ihr euch nach der Natur richten... ich wiederhole an dieser Stelle meinen Ratschlag mit dem Aufstehen und in den Wald gehen.

Auf jedenfall solltet ihr euch das ganze ab und an einmal IG (**InGame**) angucken, da die Proportionen im Editor oft nicht richtig rüberkommen.

Stehen die Bäume und seit ihr mit dem IG-Resultat zufrieden, geht es weiter.

Als nächstes solltet ihr die kleinen Details wie Steine, Pflanzen und Tiere hinzufügen. Auch hier gilt, das ihr versuchen solltet, alles möglichst gut aufeinander abzustimmen (Es sei denn ihr wollt bewusst einen 'Bruch' in der Landschaft herbeiführen... durch Abholzung beispielsweise).

Ich kann euch an dieser Stelle leider nicht viel Hilfestellung geben, weil das meiste Übung ist...

Es ist tatsächlich am besten ihr probiert einfach so viel wie möglich aus und guck was passiert ;-).

Auch solltet ihr bedenken, das evtl. Menschen mit schlechterer Hardware als deiner das ganz spielen wollen... Also lieber ein paar Bäume weniger als zuviel.

Nunja... Das ist allerdings nicht alles was man benötigt um einen stimmungsvollen Wald zu kreieren.

Im nächsten Abschnitt wenden wir uns daher der Beleuchtung und Beschallung zu, die einen nicht unwesentlichen Aspekt eines Waldes darstellt...

7.2.2: Licht & Ton

Kommen wir nun zu einem meiner Lieblingsabschnitte... Licht und Ton!

Manche Designer messen diesen Dingen viel zu wenig Bedeutung zu und stellen einfach stumpf Bäume auf - Großer Fehler!

Ich unterscheide folgende Einsatzmöglichkeiten von Licht & Ton:

Spielersteuerung

Athmosphäre

Gehen wir ersteinmal auf die Spielersteuerung ein.

Hier spielt eine ganze Menge Psychologie hinein... unter Umständen kann es nützlich sein, wenn man sich gut in Menschen hineinversetzen kann.

Nunja... wir kennen es glaub ich alle. Als Kind hat man Angst vor Dunkelheit und als Erwachsener u.U auch noch... wenn man sich im dunkeln bewegt ist man meistens vorsichtiger und reagiert schreckhafter auf Geräusche die man Tagsüber wohl irgnorieren würde.

Beim Spieler verhält sich das genauso, wenngleich die Hemmschwelle natürlich gesenkt ist...

Wir können den Spieler also beispielsweise beunruhigen wenn wir ihn durch einen dunklen Wald gehen lassen (denkt an das "Negativlicht") und dabei z.B. einige Schreie abspielen...

Wenn wir diesen Gedanken weiterverfolgen können wir o.g. so oft wiederholen, bis sich der Spieler in Sicherheit wiegt und ihn dann in einen Hinterhalt geraten lassen... Hier bieten sich enorm viele Möglichkeiten.

Neben dieser "Verhaltensteuerung" kann man auch dafür sorgen das sich der Spieler beispielsweise in eine von uns bestimmte Richtung bewegt... im Wald bietet sich hier die Möglichkeit der LeuchtPilze.

Dazu platziert man einen Pilz und ein dementsprechendes Licht (*bc_mushroom_xxx*) danebensetzt.

wenn man eine Art Pfad damit makiert wird ihm der Spieler mit an Sicherheit grenzender Wahrscheinlichkeit folgen...

Neben dem Licht kann man auch Ton einsetzen, um den Effekt zusätzlich zu verstärken.

Dazu müssen wir ersteinmal wissen wie das mit dem Ton überhaupt funktioniert...

Als erstes klicken wir dazu auf den *Sound Button* im Editor und anschließend auf *New*.

Gebt dort die ID ein und klickt anschließend rechts auf den Button *Sound Filename*. *Range* gibt die Reichweite an. Hier muss nicht zwangsläufig was eingegeben werden aber es kann nützlich sein.

(Tipp: Erstellt ein Licht mit der selben Reichweite und schaltet bei View 'Light Radius' ein, so könnt ihr sehen wie weit man den Schrei oder was auch immer hört.)

Als nächstes braucht man einen Activator mit einem Script. Am besten man benennt einfach einen bereits bestehenden um (umbedingt auf 'Yes' klicken), kopiert das Script und ändert dort einfach die ID sowie den Scriptnamen um...

(Tipp: Wenn man dem Activator das ArtFile "Editor_marker" zuweist, ist dieser IG unsichtbar.)

So nachdem wir also einen 'Sound-Emitter' unserer Wahl erstellt haben können wir ihn auch gleich einsetzen...

Um unser Beispiel von oben mit den Leuchtenen Pilzen weiterzuführen, könnte man beispielsweise einen Sound machen in dem eine Stimme 'komm zu mir' wispert (Hier bräuchte man evtl eine GetDistance Abfrage, das es ausgeschaltet wird, wenn man in Reichweite des nächsten Emitters ist).

Am Ende des mit Pilzen gekennzeichneten Weges könnte dann eine (mit magischem Blau beleuchtete) Lichtung sein, wo auf einmal von allen Seiten Stimmen wispern... Bis man von hinten angesprungen wird (die Blickrichtung kann man ja steuern indem man irgentetwas in die Mitte der Lichtung setzt...).

Nun, Spielersteuerung ist nur ein Aspekt, wie man Licht und Ton in Wäldern einsetzen kann. Der andere, und wohl am häufigsten verwendete ist das unterstreichen der Atmosphäre. Licht und Geräusche können einen sehr großen Teil dazu beitragen, das ein Wald gut wirkt. So kann man einen Tannenwald beispielsweise wesentlich unheimlicher wirken lassen, indem man dunkles Licht hineinsetzt... Evtl auch Wolfsgeheul in der Nacht oder soetwas.

Noch einmal gebe ich euch an dieser Stelle den Ratschlag, in den Wald zu gehen. Seht ihn euch genau an und versucht die Atmosphäre in euch aufzunehmen... Schließt die Augen und horcht den Geräuschen... Versucht zu erkennen welche Wirkung er auch euch hat und WARUM er diese Wirkung besitzt... Vergesst nicht, euch anzuschauen was für ein Licht in Wäldern herrscht. Ich gebe euch diesen Tipp deswegen, weil es sich schwer in Worte fassen lässt. Es gehört viel Fingerspitzengefühl und eine Menge Erfahrung dazu, Wäldern diese subtile Magie einzuhauchen, die sie austrahlen, aber man wird kläglich scheitern, wenn man sie selbst nie erlebt hat... Es ist das Zusammenspiel von Geräuschen, visuellen Eindrücken und all den kleinen Details, die man meist nur unbewusst wahrnimmt. Dieses nachzuahmen ist die wahre Kunst des Designs von Wäldern und nur sehr wenige beherrschen es wirklich...

7.3: Tipps & Tricks

7.3.1: Überhänge

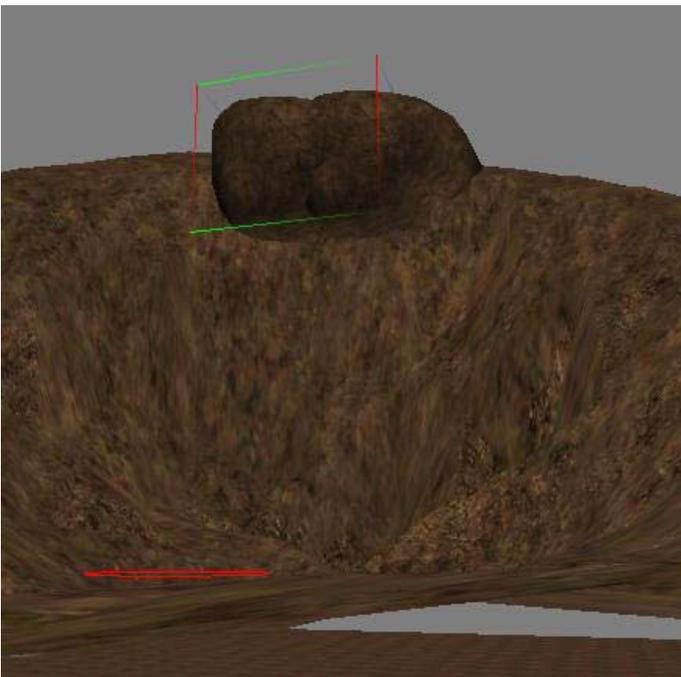
Eine Funktion, die ich beim CS gleich vermisste, ist eine Möglichkeit, Überhänge zu erschaffen. Die gibt es bis heute nicht, aber man kann sich recht gut behelfen, indem man umtexturierte Steine verwendet.

Dafür benötigen wir als erstes einmal einen Skinner, im empfehle an dieser Stelle *niftexture*.

Als nächstes suchen wir uns einen Stein (bzw. Felsen) unserer Wahl... wichtig ist hierbei das die Form zu dem Material, aus dem der Überhang bestehen soll, passt. In meinem Fall wähle ich *terrain_rock_ac_08*, da dieser relativ Flach ist...

Nun kopieren wir die .nif Datei (in diesem Fall *terrain_rock_ac_08.nif*) in einen neuen Ordner und öffnen die Kopie mit *niftexture*.

Neben der Textur können wir hier auch noch das Material bearbeiten, aber diese Funktion lassen wir am besten erst einmal aussen vor.



Ändern wir nun die Textur in eine unserer Wahl (in meinen Fall *land_default.tga*, also normale Erde) und speichern es ab.

Nachdem wir ein neues Static erstellt haben, drapieren wir unseren 'Erdklumpen' so, das ein Überhang entsteht und bearbeiten als letztes noch das Landscape etwas nach, sodass ein weicher Übergang entsteht.

Beachtet auch, das die Felsen im MW, um Polygone zu sparen, nach unten hin offen sind!

Mein Ergebnis ist links zu sehen... mit etwas Fingerspitzengefühl lässt sich sicher auch besseres erreichen.

7.3.2: Wasserfälle



Ich kann mir nicht helfen. Immer wenn ich an einen schönen, idyllischen, romantischen Wald denke, ist ein Wasserfall darin enthalten... Außerdem sieht ein Wasserfall einfach wunderbar aus und er bietet eine Fülle von Möglichkeiten, wie beispielsweise ein Eingang zu einer Höhle dahinter zu verstecken. Dabei ist es garnicht so schwer, einen Wasserfall zu erschaffen. Alle Dinge, die man dafür benötigt sind bereits im Morrowind enthalten. Nun, als erstes brauchen wir eine steile Wand. An dieser Wand drapieren wir nun Steine. Um diese Steine nun zu einem Wasserfall zu machen, setzen wir die Activators "*Ex_Vivec_Waterfall_xx*" ein. Insgesamt gibt es drei

verschiedene. Es benötigt Gedult und Übung, ehe man ein taugliches Ergebnis erreicht. Anhand meine Beispielbildes könnt ihr das Grundprinzip sehr gut erkennen.

Zum Abschluss solltet ihr noch "*Ex_Waterfall_mist*" setzen, das ist das aufgeschäumte Wasser.

Der Wasserfall ist im Original recht laut, evtl solltet ihr daher in Erwägung ziehen, die Lautstärke zu reduzieren (Natürlich nur bei einer neuen Sound und Objekt ID).

7.3.3: mehrstufiges Wasser

Gut, ich gebe zu, die Überschrift hört sich bekloppt an ;-).

Also, es gibt in MW keine direkte Möglichkeit, in Exterior Cells zwei Unterschiedliche Wasserhöhen zu haben, die Höhe ist global und immer 0.

Nun gibt es aber einen kleinen Trick, mit dem man sich behelfen kann. Wieder kommt die Lösung aus Vivec, diesmal allerdings ein Static mit dem Namen "*Ex_Vivec_P_water_01*". Wie ihr vielleicht schon vermutet handelt es sich hierbei um das Wasser, das beim Palast von Vivec zum Einsatz kommt. Das ganze hat allerdings ein paar Nachteile:

- Man kann nicht darin schwimmen
- Von unten ist es unsichtbar
- Es ist 2 Dimensional

Die ersten beiden Punkte sind sicher noch einleuchtend, letzteres hat damit zu tun, das das Palast Wasser einfach aus mehreren 2 Dimensionalen Platten mit Textur besteht.

Aus diesen Problemen ergeben sich Einschränkungen was die Nutzung anbelangt.

- Das Wasser muss so seicht sein, das der Spieler nicht merkt, das man nicht darin schwimmen kann
- Es können keine Wasserlebewesen darin schwimmen

Ansonsten ist es - soweit ich weiß - ein vollwertiger Ersatz.

7.4: Städte

Städte zu bauen ist eine sehr abwechslungsreiche Arbeit - je nach dem, wo die Stadt denn nun steht und wie sie aussehen soll.

Es ist sehr schwierig, zu diesem Thema Tipps zu geben, da es eben wirklich sehr sehr viele verschiedene Arten von Städten gibt. Daher werde ich mich in diesem Kapitel eher der richtigen Planung widmen, denn der Durchführung.

Ich kann nicht oft genug betonen, wie wichtig gute Planung für eine große Mod und insbesondere für Städte ist.

Bevor ihr den Editor auch nur anmacht, solltet ihr euch folgendes überlegen:

- Name der Stadt
- Geschichte der Stadt
- Einbindung der Stadt in TES MW
- Grundlegender Aufbau, am besten mit Skizze
- Infrastruktur (Händler, Traveling Merchants)

...und zwar am besten in dieser Reihenfolge.

Versucht möglichst, das all diese Dinge zusammenpassen. Die Geschichte wird in den meisten Fällen Auswirkungen auf den Aufbau der Stadt haben.

Ist sie schon seit Jahrhunderten an Kriegen beteiligt, wird sie höchstwahrscheinlich eine eigene Burg besitzen und vollkommen auf Verteidigung ausgerichtet sein, während eine Handelsmetropole wohl weniger Wert auf Verteidigung den auf schöne, große Häuser legen wird.

Viele Leute sparen sich die Einbindung in das TES-Universum. Aber das ist nicht unbedingt wünschenswert. Man kann wesentlich mehr Atmosphäre aufbauen, wenn man sich Mühe mit der Einbindung macht und sie nicht einfach in die Landschaft klatscht... vielleicht ein paar Gerüchte in die Welt setzen, drei, vier neue NPC's in verschiedenen Städten...

Hat man sich Gedanken darüber gemacht geht es jetzt an die konkretere Planung. Wie groß soll die Stadt sein? Wie viele Häuser? Einen Palast? Wie sollen die Wege verlaufen?

Natürlich ist das bei einem 4-Seelen-Kaff nicht notwendig, aber wenn es eine Stadt werden soll, kann man sich mit einer Skizze sehr viel Arbeit sparen...

Zum Schluss schließlich sollte man sich Gedanken über die Infrastruktur machen. Wieviele und was für Händler in der Stadt ansässig sind, hat viel mit der Geschichte zu tun.

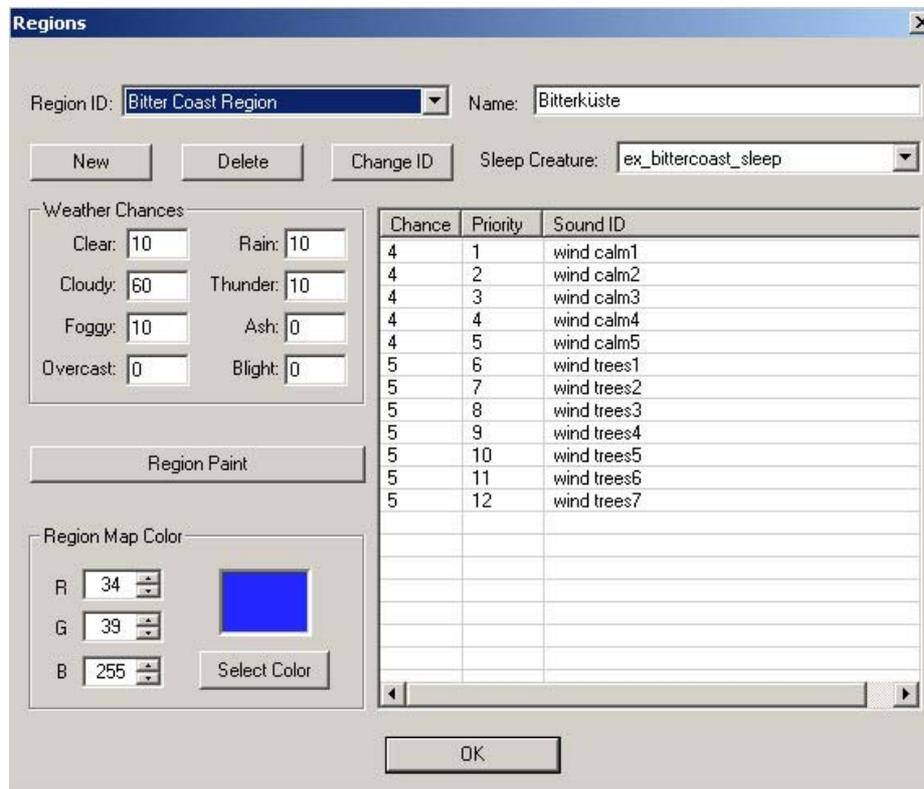
Auch über die Verkehrsanbindung sollte man sich Gedanken machen...

7.5: Region & Region Paint

Voller Bestürzen musste ich feststellen, dass dieses Kapitel in der letzten Version nicht vorhanden war. Nun, jetzt schon.

In Morrowind gibt es einige große Regionen. Sie haben alle Spezifisches Wetter, eigene Flora und Fauna, logischerweise auch einen eigenen Namen.

Dies alles kann man im *Region*-Menu abhandelt, welches sich über World->Regions öffnet.



In diesem Fenster werden die Regionen verwaltet. Rechts unter dem Namen kann man die *Leveled List* mit den Kreaturen einladen, die einen beim schlafen überraschen. Auf der linken Seite kann man das Wetter steuern, indem man Chancen (in Prozent) für die verschiedenen Wassertypen festlegt. Links daneben befindet sich schließlich die Liste mit den Sounds, die in dieser Gegend abgespielt werden soll. Man kann neue hinzufügen indem man über Gameplay->Sounds

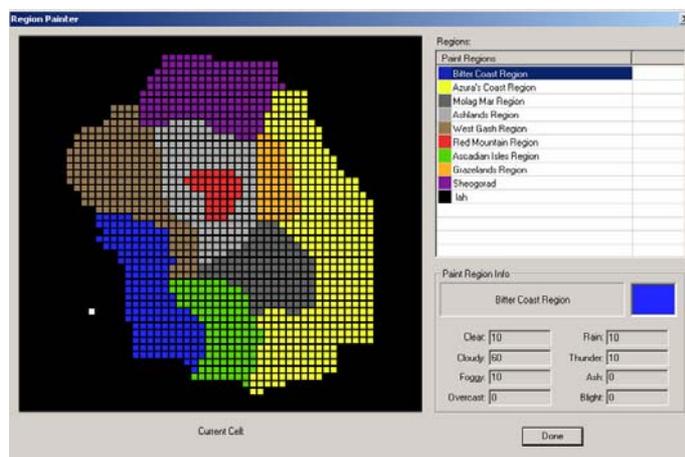
das Soundfenster öffnet und die passenden Sounds in das Regionfenster hineinzieht.

Dabei wird automatisch eine *Priority* vergeben.

Das ist wichtig, damit sich nicht mehrere Sounds überlagern.

Neben der *Priority* kann man auch eine *Chance* eingeben, die logischerweise bestimmt, mit welcher Wahrscheinlichkeit der Sound abgespielt wird.

Links unten schließlich kann man eine Farbe definieren. Es ist wichtig das diese Farbe einzigartig ist. Sie hat nicht das geringste mit dem Spiel zu tun und dient nur zu eurer Übersicht.



Durch einen Klick auf 'Region paint' gelangen wir zu diesem Fenster.

Jeder Region ist hier eine Farbe zugeordnet. Wie der Name bereits sagt, kann man hiermit Regionen 'malen', spricht man definiert ihren Verlauf und ihre Ausdehnung durch eine Farbe.

Auf diese Weise kann man bequem und einfach große Landmassen benennen.

Interiordesign für Fortgeschrittene

Nachdem wir uns etwas genauer mit dem Exteriordesign beschäftigt haben möchte ich nun gerne näher auf das Interiordesign eingehen.

Ich möchte noch einmal anmerken das ich wirklich kein In- beziehungsweise Exteriorexperte bin, sondern nur ein Multitalent das von allem etwas kann.

Aber gerne möchte ich euch meine Erfahrungen schildern, damit ihr nicht die selben Fehler macht wie ich.. Aber wie sagte mein Vater immer: *“Wer nie sein Brot im Bette aß, weiss nicht wie Krümel pieksen”* oder auf moderner: *Try and Error!*

8.1: Layout

Das Layout ist sehr wichtig für ein Interior.

Bei kleineren Interiors mit nur einem Raum ist dies selbstverständlich vernachlässigbar aber die meisten werden wohl größeres schaffen wollen.

Das ganze beginnt schon kompliziert zu werden wenn man ca. 5 Räume hat, noch viel schlimmer wird es mit der ersten Treppe.

Sollte man also so komplexe Interiors bauen wollen, sollte man sich **im Vorfeld** Gedanken über das Layout machen, damit der Spieler nicht unnötig herumirren oder weite Strecken zurücklegen muss.

Der Aufbau eines Interiors ist abhängig von vielen Faktoren. Die wichtigsten sind wohl:

- Baustil
- Gebäude/Dungeon
- Zweck des Gebäudes
- Wirkung auf den Spieler
- Inhaber (was für ein Amt hat er inne? Mag er's gern protzig? Was für eine Religion?)

Anhand dieser und weiterer Faktoren, die sie aus den oberen ergeben, kann man sich bereits ein Bild vom Aufbau des Interiors machen.

Dabei ist natürlich zu beachten ob es sich um ein Gebäude oder ein Dungeon handelt, da der Aufbau des letzterem in nicht unbedeutendem Maße von der Natur diktiert wird.

Erstmal will ich jedoch näher auf Gebäude eingehen..

Nun, einer eurer ersten Mod's wird wohl ein Haus für euren Char sein. Natürlich will man in diesem alles wichtige drin haben - viel Stauraum, einen Alchimie- und einen Trainingsraum etc pp.

Generell sollte man bei so großen Gebäuden mehrere Interiors erstellen, das spart Rechenzeit und man kann die einzelnen Interiors freier kombinieren.

Aber wie soll man sie nun anordnen?

Nun das ist abhängig von o.g. Faktoren.

Jedoch darf man nicht ausser acht lassen, das sich der Spieler schließlich darin zurechtfinden muss..

Glücklicherweise sind solche komplexen Gebäude eher die Außnahme.

Bei Dungeons verhält es sich ähnlich..

Auch hier sollte man darauf achten, dass es nicht alzu schwer ist sich schließlich darin zurechtzufinden..

Es sei denn natürlich es handelt sich um ein Labyrinth oder ähnliches.

Besonders schlimm wird es wenn es viele Türen in andere Dungeon-Interiors gibt die wiederum in Dungeon-Interiors verzweigen..

8.2: Atmosphäre

Die Atmosphäre ist ein sehr wichtiger Bestandteil jedes In und Exteriors. Im letzten Kapitel haben wir uns ja bereits mit Licht und Ton beschäftigt.

Das werden wir auch hier wieder tun.

Aber erstmal wenden wir uns einem anderen, ebenfalls wichtigen Aspekt zu.

Dekoration..

8.2.1: Deko

Am Anfang des Tutorials habe ich die Dekoration nur kurz angesprochen, dabei ist sie äußerst wichtig. Nichts ist schlimmer als in ein Haus zu kommen in dem angeblich ein NPC lebt und einen sogut wie leeren Raum vorzufinden, wo lieblos ein paar Flaschen hingeklatscht wurden..

Das Dekorieren eines Interiors ist mir der aufwendigste Teil des Interiordesigns..

Das wichtigste was man für eine gute Dekoration benötigt ist Gedult. Man muss sich schlichtweg Zeit lassen um die ganzen Kleinigkeiten zu schaffen, die einen Großteil der Atmosphäre schaffen.

Dementsprechend kann ich euch nur einige Tipps und Anregungen mit auf den Weg geben.

Es sollte klar sein das die Dekoration passend gewählt sein sollte.. In einer kleinen Hütte mir nur einem Zimmer findet wohl kaum ein großes Doppelbett platz..

Auch Regale wird sich der Besitzer einer solchen Hütte eher weniger leisten können, d.h. wird er seine Habe in Säcken oder Körben lagern. Das meiste davon dürften dann auch Naturalien sein.

Ein reicher Geschäftsmann hingegen kann sich vielleicht sogar Silberbesteck leisten..

Desweiteren sollte die Deko zum NPC passen. Ein Alkoholiker beispielsweise wird wohl eine Menge leerer Flaschen herumstehen haben, bei einem Jäger werden Pfeile, Schnitzwerkzeug und Felle herumliegen und ein Handwerker schließlich größtenteils Nahrungsmittel und eine kleine Schatulle mit seinen Ersparnissen..

Auch hier ist es am besten wenn ihr euch die MW Interiors und vielleicht sogar mal das wirkliche Leben anschaut um ein Gefühl dafür zu bekommen.

8.2.2: Beleuchtung&Sound

Im Abschnitt 7.2.2 habe ich bereits über die Einsatzmöglichkeiten von Licht und Ton gesprochen, in Interiors verhält sich das alles jedoch ein wenig anders.

Zum einen ist der Raum meist schlichtweg kleiner, zum anderen gibt es in Interiors ganz andere Einsatzgebiete.

Ersteinmal ein wichtiger Hinweis: Licht stört sich in MW nicht an Hindernissen, sprich es leuchtet durch Wände hindurch.. Besonders nervig ist dies in dem Stockwerk darüber bzw darunter.. Daher sollte man sich überlegen ob man die anderen Stockwerke entweder höher bzw. tiefer liegt oder gar in ein ganz anderes Interior packen..

Für den Einsatz von Sound sei gesagt, das man die Reichweite aufgrund der geringeren Entfernungen entsprechend verkleinern sollte.

Nundenn wo ergeben sich hier gute Einsatzgebiete?

Als erstes natürlich um die Atmosphäre zu unterstreichen.

Hier gibt es eigentlich nicht als zuviel zu sagen..

In Ahnengrüften hört man auch in MW sogut wie immer die Geister herumwuseln und selbstverständlich sind auch einmalige Schreie etc möglich.

Auch hier sollte man einfach mal ausprobieren um ein Gefühl für (in erster Linie) das Licht zu bekommen.

Wie ihr seht wieder eine Sache wo man nicht wirklich viel zu sagen kann, also gehen wir gleich weiter.

8.3: Tipps & Tricks

Hier werde ich Tipps 'n Tricks von verschiedenen Moddern zusammenstellen die ich so im netz finde. Bisher leider nur einer aber ich bin zuversichtlich das es in zukünftigen Versionen mehr sein werden.

8.3.1: Galerien

thx to XenArien



Vielleicht habt ihr euch schon hin und wieder gedacht, wie nett doch eine Galerie wäre. XenArien hat dazu im Offiziellen Editor Forum seinen Vorschlag gepostet und mir freundlicherweise gestattet, ihn hier miteinzubringen.

Er ist nur für die *c_plain* - Teile, lässt sich denke ich aber auch gut für andere Stile adaptieren.

Wie es funktioniert sieht man rechts.

Man baut einen Raum im *c_plain* Stil, jedoch ohne die Center Teile.

Das setzt man dann nocheinmal direkt darunter.

Nun nimmt man *c_plain_room_center* Teile und vergrößert sie mittels 3D Scale auf 2.0. Diese kommen dann natürlich in die Mitte.

Dann nimmt man *Wooden_Pillar* als Stütze für die Oberseite. Für die Unterseite das gleiche, jedoch um 180 grad um die Z-Achse gedreht und mit 3D Scale 1.2.

Die leere Stelle zwischen den beiden Stockwerken füllt man mit *ex_de_shack_plank_4* auf 3D Scale 2. Dann nurnoch ein *furn_com_fence* In dir obere Etage und es sollte so Aussehen wie rechts.



Wie in den meisten RPG's gibt es auch in Morrowind eine Vielzahl von Rassen mit den unterschiedlichsten Eigenschaften.

Nun kann es sein, das man selbst eine Rasse in das Spiel einbringen will. Dies kann eine spielbare Rasse sein, muss es aber nicht. Auch kann man Rassen erschaffen, um bestimmte Dinge zu realisieren wie z.B. eine Trainingspuppe oder ähnliches.

Beim erstellen von Rassen gibt es einige wichtige Dinge zu beachten, die ich in diesem Kapitel näher erläutern möchte.

9.1: Das Rassenfenster

Als erstes wollen wir einmal das Fenster, indem die Rasse erstellt wird näher betrachten. Geöffnet wird es mit einem Klick auf *Character* -> *Race*.

Daraufhin sehen wir folgendes vor uns:

Base Attributes		
	Male	Female
Str	45	45
Int	30	40
Wil	50	45
Agi	35	35
Spd	30	30
End	50	50
Per	30	25
Luc	40	40
	310	310

Spell	Type
SP orc_beserk	Eigensch...
SP resist magicka_25	Fähigkeiten

ID: Hier wird die ID der Rasse angezeigt. Durch einen Klick auf *New* kann man eine neue hinzufügen und mit *Delete* wieder löschen. (Hinweis: Kommt nicht auf die Idee, eine bestehende Rasse zu löschen.)

Name: Der Name der Rasse

Skill Bonus: Die Skillboni der Rasse. Es können maximal sieben gewählt werden.

Base Attributes: Hier kann man die Attribute der Rasse bestimmen.

Specials: Hier kann man die Fähigkeiten und Eigenschaften wählen, die einem Angehörigen dieser Rasse von vornherein zu Verfügung stehen

Description: Die Beschreibung der Klasse. Unnötig wenn nicht spielbar

Height/Weight: Die Größe und das Gewicht. Letzteres bestimmt u.A. die Sprunghöhe.

Playable: Bestimmt, ob die Klasse spielbar, d.h. vom Spieler wählbar ist

Beast Race: Ist es aktiviert, können keine Stiefel getragen werden.

9.2: Erstellen einer neuen Rasse

9.2.1: Konzeption

Die Konzeption einer neuen Rasse ist enorm wichtig (es sei denn es handelt sich um rein "technische" Rassen wie z.B. Trainingsdummies). Ihr solltet bedenken, das man sie nicht so einfach in eine bestehende Welt hineinklatschen kann. Sie benötigen eine eigene Geschichte, eine eigene Kultur, dementsprechende Attribute sowie Fähigkeiten und in den meisten Fällen auch ein anderes Aussehen. Daher sollte man sich im Vorfeld Gedanken darüber machen was man genau will und - vielleicht sogar wichtiger - wie das entstanden sein könnte, denn die Rasse braucht ja eine Geschichte.

Als nächstes solltet ihr euch die speziellen Fähigkeiten eurer Rasse überlegen.. Vielleicht Nachtsicht? Oder ein Berserkerartiger Blutausch?

Auch die Attribute und die Fähigkeitsboni sollten wohl überlegt sein.. Hier sollte man sich bei spielbaren Rassen schon an die MW Vorgaben von 45 Pkt Fertigkeitboni und 310Pkt bei den Attributen halten um die Balance nicht zu stark zu gefährden..

Hat man sich das alles überlegt gehts ans eingemachte.

9.2.2: Fähigkeiten und Eigenschaften

Nachdem die Attribute und Fertigkeitboni festgelegt wurden, gehts an die speziellen *Fertigkeiten* und *Eigenschaften*. Technisch gesehen sind dies Zauber.

Hierbei ist die Übersetzung irreführend. Eine *Fähigkeit* wirkt ständig, während eine *Eigenschaft* ein Zauber ist, der immer funktioniert und den man nur 1x pro Tag einsetzen kann.

Um diese zu kreieren gehen wir in den Reiter Spellmaking, machen einen neuen Spell, wählen dort die Effekte und schließlich unten links *Eigenschaft* bzw *Fähigkeit*.

Diese werden dann einfach rechts oben in das Rassenfenster gezogen.

Auch hier sollte man darauf achten das man keine zu starken Effekte wählt, sodenn es sich um eine spielbare Rasse handelt.

9.2.3: Bodyparts

Damit ihr eure Rasse nun verwenden könnt, müsst ihr ihre Bodyparts definieren.

Die Bodyparts, die zu Rassen gehören haben folgendes Namensschema: *B_N_Rassenname_M/F_Bodypartbezeichnung_xx*

Dabei steht *M/F* für *Male/Female*, also M wenn es ein männliches Bodypart ist und F für ein weibliches..

Um diese nun für eure Rasse zu definieren wählt als Typ *Skin* und bei *Race* eure Rasse.

Was ihr alles braucht seht ihr anhand der bereits vorhandenen Rassen - generell spart es Tipparbeit, wenn ihr einfach Teile einer bereits existierenden Rasse umbenennt und ihnen ggf. ein neues Bodypart zuweist.

Damit umgeht man auch den sog. "1st Hand Bug" bei dem die Hände in der 1st Person Ansicht nicht angezeigt werden. Falls es doch einmal der Fall sein sollte hier ein Hinweis: Die ID der 1st Hand muss **exakt** die gleiche sein wie die der 'normalen' Hand, mit dem Unterschied, das ein ".*1st*" angehängt ist.

Dropdown Gameplay & Character

10.1: Gamesettings

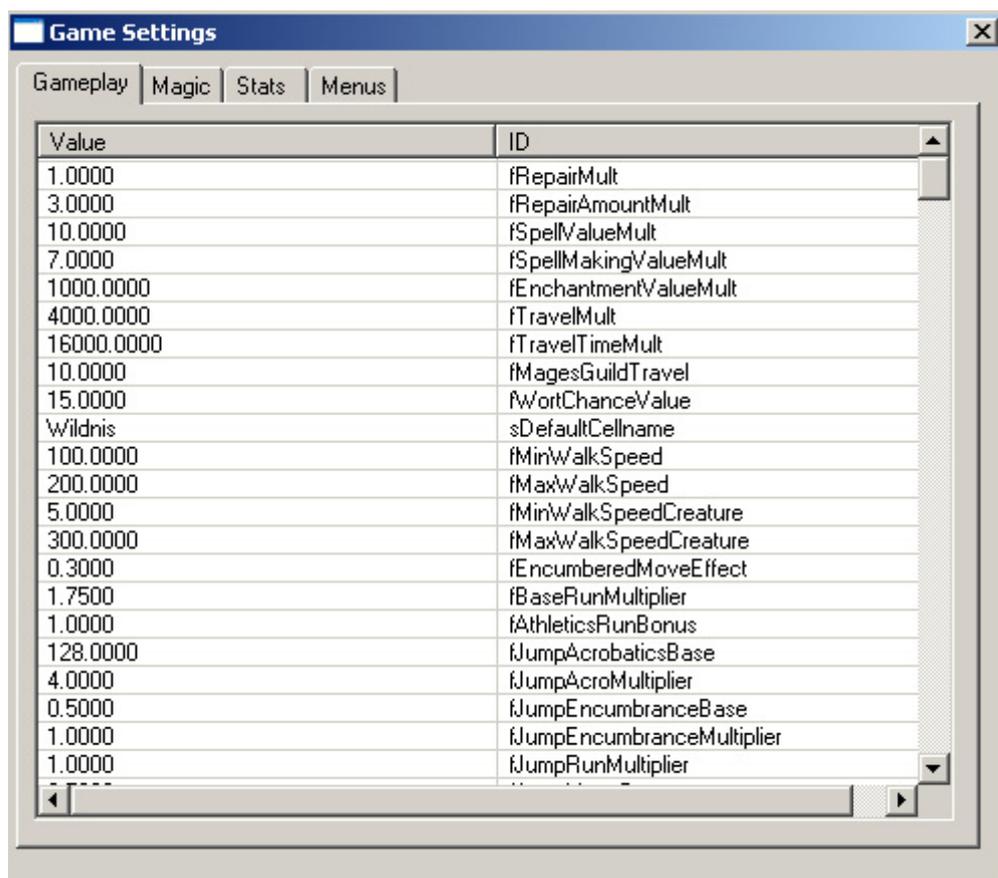
Nun wird es Zeit, das wir etwas tiefer in den Editor hineintauchen und quasi hinter die Kulissen sehen. Die *Game Settings* sind ein enorm mächtiges Werkzeug - in ihr sind Dinge gespeichert wie z.B. die Sprünghöhenveränderung, die Acrobatics mit sich bringt. Oder ab wann man 3 Pkt bei seinen Attributen verteilen kann. Oder wie schnell die Zeit vergeht.

Das heisst man kann direkte Kontrolle über das gesammte Tuning des Spiels erreichen und so ziemlich alles ändern, was man ändern möchte..

Beginnen wir also direkt

10.1.1: Das Game Settings Fenster

Das Game Settings Fenster wird über Gameplay -> Settings geöffnet und sieht so aus:



Über den Kateireiter am oberen Rand können wir die verschiedenen Unterkategorien auswählen die da im einzelnen sind: *Gameplay* - Hier stehen allgemeine Dinge wie z.B. Sprunghöhe oder Skillgain. *Magic* - Hier stehen Einstellungen für magische Effekte - auch die Namen kann man hier verändern *Stats* - Wie *Magic* nur für Fähigkeiten. *Menus* - Hier stehen z.B. die

Namen der Monate oder auch einfach die Beschriftung der Schaltflächen wie "OK", "Weiter", etc.

Wollen wir nun einen Wert ändern klicken wir ihn einfach an, als ob wir ihn umbenennen wollten..

10.1.2: Anwendungsgebiete

Mit den Game Settings lassen sich eine Menge Dinge realisieren die sonst unmöglich wären.

Auch wenn sie auf den ersten Blick unscheinbar wirken, kann man mit ihnen das Spiel ganz an seine Wünsche anpassen..

Nun, wo ergeben sich hier Anwendungsgebiete?

Eigentlich überall.. Beispielsweise ist ein Mod möglich, der dafür sorgt, das die IG Zeit genauso schnell vergeht wie die RL Zeit, was sehr nützlich für Vampire wäre..

Man kann auch dafür sorgen das man nichtmehr so schnell hohe Multiplikatoren bekommt oder dafür Sorgen das die Beliebtheit mit dem Bekanntheitsgrad schneller steigt..

Noch viel wichtiger werden die GameSettings, wenn man eine TC, eine Total Conversion, d.h. eine komplett neue Welt erschaffen will.

Natürlich wird man in den meisten Fällen versuchen, diese TC so eigenständig wie möglich zu machen und dazu gehören auch eine Menge GameSettings..

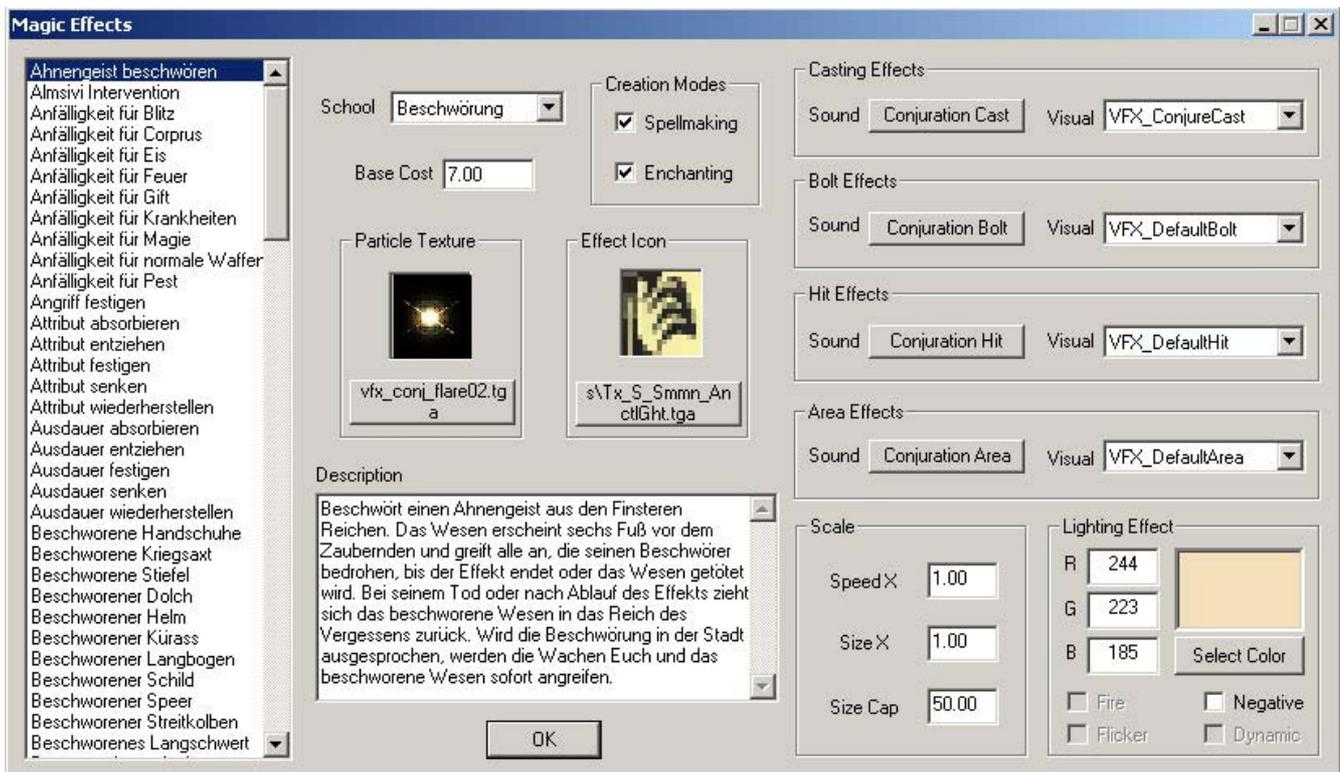
Nun, wieder ein Kapitel in dem nicht viel zu sagen ist, wobei die GameSettings einer der mächtigsten Teile des Construction Set sind.

Es sei an dieser Stelle noch darauf hingewiesen, das man keine GameSettings verändern sollte, wenn man dies nicht wirklich will, da jede Änderung schwerwiegende Konsequenzen haben kann.

10.2: Magic Effects

Öffnen wir über das Dropdown *Gameplay* das "Magic Effects" Fenster, so sehen wir folgendes vor

uns:



In diesem kompliziert wirkenden Fenster kann man nicht weniger machen, als die Magischen Effekte, wie z.B. die Partikel, die beim zaubern entstehen zu verändern.

Ganz Links sind alle Zauber und Effekte aufgelistet.

Rechts daneben kann man wählen, welcher Zauberschule der Zauber angehört und wie hoch seine

minimalen Manakosten sind.

Rechts daneben kann man wählen, ob sie beim *Entchanting* (Verzaubern) oder *Spellmaking* (Kreieren v. Zaubern) vorkommen können.

Darunter wird es schließlich Interessant: *Particle Textur* bestimmt Farbe und Form der Partikel, *Effect Icon* das Bild, das im Spiel dann z.B. im Zauberfenster zu sehen ist.

Description ist schließlich die Beschreibung.

Auf der rechten Seite wird es ein wenig kompliziert.

In den vier Abschnitten *Casting Effects*, *Bolt Effects*, *Hit Effects* und *Area Effects* kann man sowohl den Sound als auch das Aussehen bestimmen.

Für ersteres reicht ein klick auf den Button neben Sound. Letzteres wählt man in dem Dropdown-Menu daneben.

Ganz unten schließlich kann man einmal die Geschwindigkeit (*Speed X*), Größe (*Size X*) sowie Maximalgröße (*Size Cap*) einstellen.

Rechts daneben lässt sich schließlich die Farbe des Lichtes einstellen, wobei weiß unsichtbar bedeutet.

10.3: Skills

Über Character -> Skills gelangen wir zu folgendem Fenster:

Skill	Actions	Use Value
Akrobatik	Jump	0.15
Governing Attribute	Fall	3.00
Stärke		
Specialization		
Diebeskünste		

Description

Diese Fertigkeit ermöglicht es, über weite Entfernungen zu springen und beim Fallen aus großen Höhen Verletzungen zu vermeiden. Versierte Akrobaten können dahin gelangen, wo andere keinen Zugang haben, und sind außerdem in der Lage, während des freien Falls die Richtung zu ändern.

OK

In diesem Fenster kann man - wie vermutet - die Skills verändern.

Genauer gesagt, einmal das zugehörige Attribut 'Governing Attribute', Die Spezialisierung, zu der dieser Skill gehört, die dazugehörige Beschreibung, sowie, und das ist wohl am interessantesten, den Faktor mit dem es das IG-Spielgeschehen beeinflusst. Im Falle von Akrobatik sind das 'Jump' und 'Fall', also Sprungweite sowie -höhe und den Fallschaden. Je höher der Dazugehörige Wert 'Value', desto größer logischerweise der Effekt.

Mit diesen Einstellungen lassen sich ganz gewaltige Balancing-Veränderungen vornehmen.. Ohne Rüstung zu schwach? Kein Problem, erhöhen wir einfach den Faktor..

10.4: Class

Über Character -> Class erreichen wir folgendes Fenster:

Wie der Name unschwer erkennen lässt, lassen sich hier neue Klassen erstellen und die bereits vorhandenen ändern. Das ist denkbar einfach in denen man die Haupt- (*main*) und Nebenskills (*Minor*) sowie die Attribute (*Primary Attributes*) und die Spezialisierung. Soll die Klasse vom Spieler spielbar sein, so macht man einfach ein Häkchen bei 'Playable', in diesen Fall sollte man rechts daneben auch eine Beschreibung schreiben.

Soll die Klasse *nicht* spielbar sein, sondern nur für NPC' gelten, kann man unten die *Services* auswählen, die diese Klasse standartmäßig anbietet.

Die Klasse *Schmied* beispielsweise sollte standartmäßig ein Häkchen bei 'Weapons' sowie rechts bei 'Repair' haben.

Damit kann man dieses Kapitel auch gewohnt schnell abschließen und wir kommen zum nächsten Thema.

10.5: Birthsigns

Birthsigns heißt 'Sternzeichen' und gibt eine Menge Möglichkeiten.

Auch hier ist das Fenster denkbar einfach aufgebaut.

Rechts kann man eine ID festlegen, darunter den Namen und das Bild sowie die Beschreibung angeben.

Auf der rechten Seite kann man wie auch im *Race*-Fenster Zaubersprüche hineinziehen, die diesem Sternzeichen eigen sind.

Die Bilder sollten sich im Ordner Textures/ Birthsigns befinden und die Abmessung von 256x128 haben.

Viel mehr gibt es hier nicht zu beachten.

10.6 Faction

Vielleicht willst du ja eine neue Gilde erstellen.

In diesem Fall solltest du, nachdem du ein Kontzept hast, erstmal das *Faction*-Fenster öffnen.

Rank Name	Attrib 1	Attrib 2	Pri Skill	Fav Skill	Fact Rep
Clanfreund	30	30	0	0	0
Freund	40	30	30	20	5
Bruder	45	35	40	20	10
Geweihter	50	40	50	25	20
Clansmann	55	45	60	25	30
Führer	60	50	70	30	45

Faction	Reaction
Ashlanders	3
Sixth House	-3
Thieves Guild	-2
Imperial Legion	-2
Fighters Guild	-2
Clan Quarra	-3
Clan Berne	-3
Clan Aundae	-3
Camonna Tong	-1

Das sieht Komplizierter aus als es eigentlich ist. Über 'Hidden from PC' kann man eine Fraktion erstellen, die für den Spieler verborgen ist, wie das z.B. bei *TwinLamps* der Fall ist.

Darunter kann man die verschiedenen Stufen festlegen, eine neue erstellt ihr einfach, indem ihr auf ein leeres Feld klickt.

Für jeden Rang müssen die Werte der beiden Hauptattribute manuell eingestellt werden, die der Skills werden

automatisch berechnet.

Unten kann man schließlich die Reaktionen anderer Gilden auf diese Gilde einstellen. Telvanni beispielsweise sind bei den Redoran nicht sonderlich beliebt etc..

Um eine neue Reaktion zu erstellen, einfach Rechtsklick->New. Rechts kann man dann die Gilde sowie die Reaktion einstellen.

Das ist jedoch nur der erste Schritt zur eigenen Gilde.

Als nächstes brauchen wir zumindest einen Gildenchef, der Aufgaben verteilt und Beförderungen vornimmt.

Sehen wir uns das einmal am Fall von Trebonius, dem Erzmagier der Magiergilde an.

Hier ein Ausschnitt aus dem Topic 'Beförderung'. Es gibt hier zwei wichtige Faktoren. Zum einen ist es die *Function* 'Rank Requirement'. Diese kennt drei Zustände: 0, 1, 2 und 3. 0 bedeutet, man hat weder genug Aufgaben erfüllt, noch sind die Skills hoch genug. Bei 1 hat man nicht genug Aufgaben erfüllt,

2 definiert einen Mangel an Erfahrung.

Bei 3 schließlich stimmt alles und die Beförderung kann beginnen.

Dazu muss in das Result Fenster folgendes eingetragen werden:

PCRaiseRank "GuildID"

Damit wird der Spieler dann befördert.

Die Anzahl der erfüllten Aufgaben werden über '*FactRep*' abgehandelt, die man auch im Faciton-Fenster verstellen kann.

Verschiedenes

In diesem Kapitel werde ich Tutorials von anderen Moddern, wie z.B. Iladrions TESAME Tutorial, das er mir freundlicherweise zur Verfügung gestellt hat, sollte es denn einmal fertig werden, sowie anderen Kram, der sonst nirgendwo hineinpasst, aber dennoch wichtig ist, sammeln.

11.1: Die Morrowind.ini

Die *.ini Dateien gehören zu den mächtigsten Werkzeugen der Spiele. In ihr sind eine Menge Einstellungen gespeichert.. Im Falle von MW ist die *Morrowind.ini* quasi ein erweitertes GameSettings. Allerdings ist sie nur bedingt für Mods zu empfehlen da in ihr auch Dinge wie die Tastenbelegung, Sichtweite, etc gespeichert werden. Daher ist es eigentlich nur für den privaten Gebrauch oder für TC's zu empfehlen, wobei bei letzterem vorher eine Sicherheitskopie angefertigt werden sollte - vom Endnutzer selber oder vom Installationsprogramm.

Nun, im folgenden werde ich alle Einstellungen der Morrowind.ini, die für uns von Interesse sein könnten hier auflisten. An dieser Stelle möchte ich bemerken das die Liste selbstverständlich noch erweitert wird und in dieser Version keineswegs vollständig ist.

Alle Angaben die mit einem ';' beginnen sind Kommentare und werden nicht verarbeitet.

Die Angaben in Klammern sind die Standartwerte. '1' steht für ja und '0' für nein alles andere sind konkrete Werte. Der Wert muss ohne Freizeichen hinter einem Gleichzeichen stehen, um erkannt zu werden.

[General]

Show FPS (0)	- Bestimmt, ob die Framerate dauerhaft angezeigt wird
Max FPS (240)	- Bestimmt die maximale Framerate. Sollte auf Frequenz des Monitors gestellt werden.
TryArchiveFirst (0)	- Wenn 1, schaut das Spiel erst in den bsa Achriven nach einer Datei.
SkipProgramFlows (0)	- ?
DontThreadLoad(0)	- ?
ThreadPriority (-1)	- ?
ThreadSleepTime (4)	- ?
Clip One To One Float (1)	- ?
Flip Control Y (0)	- Bestimmt ob die Kontrollen an der Y gespiegelt werden (Rechts=Links)
SkipKFExtraction (1)	- ?
PC Footstep Volume (0.7)	- Bestimmt die Lautstärke der Schritte
Disable Audio(0)	- Hier kann man den Ton deaktivieren
Background Keyboard (0)	- ?
Use Joystick (0)	- ?
Joystick X Turns (0)	- ?
Joystick Look Up/Down (6)	- ?
Joystick Look Left/Right (3)	- ?
Create Maps Enable (0)	- Wenn 2 kann man mit dem Konsolenbefehl <i>CreateMaps</i> [Filename.esp] Karten im bmp Format erstellen
Screen Shot Enable (0)	- Bestimmt, ob man Screenshots machen kann
Screen Shot Base Name (ScreenShot)	- Der Name der Screenshots
Screen Shot Index (0)	- Die höchstmögliche Zahl hinter den Screenshotnamen

Beta Comment File ()	-?
Interior Cell Buffer (10)	- Bestimmt, wieviel Interiorzellen im Speicher verbleiben
Exterior Cell Buffer (32)	- Dasselbe für Exteriorzellen
Number of Shadows (6)	- Anzahl der maximalen Schatten?
Maximum Shadows Per Object (1)	- Anzahl der Schatten für ein Objekt?
AllowMultipleEditors (0)	- Wenn ja, kann man mehrere Editoren starten
ExportDialogueWithHyperlinks (0)	-?
MaintainImportedDialogueOrder (1)	-?
KeepImportedDialogueIDs (1)	-?
DisableLanguageCheck (0)	- Hat man EV Savegames die man mit der DV öffnen will muss hier 1 stehen
UseExistingTempFile (0)	-?
CanMoveInfosWhileFiltered (0)	-?
Editor Starting Cell (0)	- Hier kann man eine Cell definieren, an der der Editor startet
Editor Starting Pos (0.000000,0.000000,16384.000000)	- Hier steht die Standart Kamera Position des Editors
Editor Starting Dir 0 (1.000000,0.000000,0.000000)	- Dasselbe für die Blickrichtung?
Editor Starting Dir 1 (0.000000,-0.000000,1.000000)	- s.o.
Editor Starting Dir 2 (0.000000,-1.000000,-0.000000)	- s.o.
Subtitles (0)	-?
ShowHitFader (0)	-?
Language (jeweilige Sprache)	- Die Sprache des Spiels
[LightAttenuation]	
UseConstant (0)	-?
ConstantValue (0.0)	-?
;	
UseLinear (1)	- Ich nehme an eine Rendermethode für die Lichtberechnung
LinearMethod (1)	- und diverse Einstellungen dafür
LinearValue (3)	- s.o.
LinearRadiusMult (1.0)	- s.o.
;	
UseQuadratic (0)	- Eine weitere Rendermethode für Licht?
QuadraticMethod (2)	- und die Einstellungen
QuadraticValue (16.0)	- s.o.
QuadraticRadiusMult (1.0)	- s.o.
;	
OutQuadInLin=0	-?
[Inventory]	- ? Lasse es raus
[Map]	
Travel Siltstrider Red (238)	- Rotanteile der Farbe für die Siltstrider - Linien auf der Karte
Travel Siltstrider Green (230)	- Grünanteile
Travel Siltstrider Blue (30)	- Blauanteile
Travel Boat Red (30)	- s.o. nur für's Schiff
Travel Boat Green (30)	- s.o.
Travel Boat Blue (230)	- s.o.
Travel Magic Red (230)	- s.o. diesmla für die Magier
Travel Magic Green (30)	- s.o.
Travel Magic Blue (30)	- s.o.
Show Travel Lines (0)	- wenn aktiviert, werden die Verbindungen in den jeweiligen oben definierten Farben auf der Karte angezeigt
[MenuStates]	- ? Ebenfalls rausgelassen
[CopyProtectionStrings]	
No CDROM Message (CD-ROM/DVD-Laufwerk wurde nicht gefunden.)	- Hier kann man die Nachricht bestimmen, die angezeigt wird, sollte kein Laufwerk gefunden werden.

No CDROM Title (CD-ROM-Laufwerk wurde nicht gefunden)	- Der Titel der Fehlermeldung
Enter CD Message (Legen Sie die Morrowind-CD in ein beliebiges CD-ROM/DVD-Laufwerk ein.)	- Fehlermeldung, die angezeigt wird wenn keine MW CD gefunden wurde
Enter CD Title Message (Morrowind-CD nicht gefunden)	- Titel der Fehlermeldung
[GeneralWarnings]	- selbsterklärend
Yes (Ja)	- ebenfalls
No (Nein)	- Dieser Text wird angezeigt wenn es einen Fehler gegeben und man die wahl hat das Programm zu verlassen oder fortzufahren
Continue (Ausführung des Programms fortsetzen?)	- Die Fehlermeldung kommt, sollte man ein Savegame laden und ein Plugin wurde während der letzten Speicherung verändert.
MasterMismatchWarning (Eine der Dateien, die für "%s" benötigt werden, wurde seit der letzten Speicherung verändert. Dies kann zu Fehlern führen. Nach erneutem Speichern wird diese Meldung nicht mehr erscheinen. Dadurch ist aber nicht notwendigerweise der Fehler behoben.	- Diese Fehlermeldung kommt wenn ein Plugin eine andere Masterdatei benötigt als die vorhandene (z.B. wenn man EV Plugins in der DV starten will)
GeneralMasterMismatchWarning=Die benötigte Version der Master-Datei konnte von einem oder mehreren Plugins nicht gefunden werden. Während des Ladens oder Spielens können daher Fehler auftreten. Mehr Informationen hierzu finden Sie in der Datei "Warnings.txt".)	
[Movies]	- Hier kann man z.B. das Intro ändern..
Company Logo (bethesda logo.bik)	- Das Logo von Bethesda
Morrowind Logo (mw_logo.bik)	- Das MW Logo
New Game (mw_intro.bik)	- Das eigentliche Intro wenn man ein neues Spiel startet
Loading (mw_load.bik)	- ?
Options Menu (mw_menu.bik)	- Die Menuanimation
[Question 1-10]	- Hier kann man die Chargenfragen editieren.
[Water]	- Einstellungen die das Wasser betreffen
[PixelWater]	- Das selbe für das Pixelshading Wasser
[Blood]	
Model 0 (BloodSplat.nif)	- Hier kann man das Model für das 'normale' Blut verändern
Model 1 (BloodSplat2.nif)	- dasselbe für die weißen Splitter des Skelletes
Model 2 (BloodSplat3.nif)	- und für die goldenen Funken
Texture 0 (Tx_Blood.tga)	- Hier stehen die Texturen desselben
Texture 1 (Tx_Blood_White.tga)	- s.o.
Texture 2 (Tx_Blood_Gold.tga)	- s.o.
Texture Name 0 (Default (Red))	- und die Namen
Texture Name 1 (Skeleton (White))	- s.o.
Texture Name 2 (Metal Sparks (Gold))	- s.o.
[PreLoad]	
Cell 0 (Kaiserliches Gefangenenschiff)	- Hier kann man einstellen, welche Zellen bei einem neuen Spielstart von 'vorgeladen' werden sollen
Cell 1 (Seyda Neen)	
	- ?

[Moons]	- Einstellungen zu den Monden
Masser Size (94)	- Größe von Masser
Masser Fade In Start (14)	- ?
Masser Fade In Finish (15)	- ?
Masser Fade Out Start (7)	- ?
Masser Fade Out Finish (10)	- ?
Masser Axis Offset (35)	- ?
Masser Speed (.5)	- Die Geschwindigkeit mit der sich Masser bewegt
Masser Daily Increment (1)	- ?
Masser Fade Start Angle (50)	- ?
Masser Fade End Angle (40)	- ?
Masser Moon Shadow Early Fade Angle (0.5)	- ?
Secunda Size (40)	- Secundas Größe
Secunda Fade In Start (14)	- ?
Secunda Fade In Finish (15)	- ?
Secunda Fade Out Start (7)	- ?
Secunda Fade Out Finish (10)	- ?
Secunda Axis Offset (50)	- ?
Secunda Speed (.69)	- Die Geschwindigkeit mit der sich Secunda bewegt
Secunda Daily Increment (1.2)	- ?
Secunda Fade Start Angle (50)	- ?
Secunda Fade End Angle (30)	- ?
Secunda Moon Shadow Early Fade Angle (0.5)	- ?
[Weather]	- Einstellungen die das Wetter betreffen
[Weather XX]	- Einstellungen zu den einzelnen Wetterlagen - größtenteils Farben
[Fonts]	- Hier kann man die Schriftarten definieren die IG vorkommen sollen
[FontColor]	- und hier deren Farben
[Cursors]	- Hier kann man die einzelnen Cursor's definieren
[Editor]	- Hier steht, welche Fenster mit dem Start des Editors geöffnet werden sollen
Open Render (1)	
Open Cell View (1)	
Open Object Window (1)	
Open Preview (0)	
[WhoCanMerge]	
acheng (1)	- Wenn man hier <i>PCNAME=1</i> (PCName steht in der Programmflow.txt in der zweiten Zeile) kann man mit dem TESCS esp zu esm Dateien konvertieren - Hierbei wird jedoch die esp gelöscht! (Thx to LordKain)
[Level Up]	- Hier stehen die Level-Up-Texte
AllowYesToAll ()	- (standardmäßig nicht vorhanden) wenn 1, kann man alle Fehlermeldungen mit einem Klick erledigen.

12.2.: Übersetzung von EV MOD's

(thx to hochbass)

Obwohl ich das schon ne Weile mache lerne auch ich nie aus.

Dank auch an Iladrion , der mich auf Möglichkeiten hingewiesen hat, auf die ich allein nur schwer gekommen wäre.

Umsetzung DV nach EV -Mods

0. Die Mod als aktive Datei im TESCS laden und kontrollieren was überhaupt drin ist.

Neue geänderte ID Einträge sind mit einem "*" beim ITEM-COUNT versehen.

Hier entscheidet sich nun der Aufwand für eine Umsetzung.

Was auch immer in die Spielwelt eingefügt wurde, muß irgendwo mit dem Original verknüpft werden. Das muß herausgearbeitet werden.

Für die Umsetzung ist es wichtig zu wissen, das alle Objekte , NPC etc immer mit Ihrer ID benutzt und identifiziert werden außer Verweise auf CELLS. Bei CELLS wird der Namen direkt verwendet, weil diese keine ID besitzen. Und die DV- und EV- Namen sind unterschiedlich. Das ist eigentlich eine der Hauptursachen warum EV-Mods nicht laufen.

Ich unterscheide mal verschiedene Stufen des Aufwandes

Stufe 1: Mods, die nichts an der Spielwelt hinzufügen

sondern nur Änderungen an definierten Items in den Parameterlisten durchführen wie NPCReplace , dynamicregenmagica oder ähnlichen brauchen meist nicht übersetzt werden und laufen auch mit der DV.

Stufe 2: Die Mod fügt nur Items in neue Interior-Cells hinzu. Es werden keine NPC, Quests etc erstellt. (z.B. Haus-Mod's etc.)

In diesen Mods muß man eventuell nur die Einstiegszelle anpassen.

Wenn die Tür zum neuen Haus in einer Außenzelle steht, braucht man nichts machen.

Wenn das Teleportammulett oder die Tür zum neuen Haus in einer internen Zelle mit EV-Namen steht bzw. liegt, dann muß alles was sich geändert hat in die gleiche Zelle mit DV-Namen , positionsgenau übertragen werden.

Dafür gibt es im Prinzip zwei Möglichkeiten:

1. geändertes im Renderfenster markieren und mit Strg.-C kopieren. Im Renderfenster die DV-Zelle öffnen und aktivieren (oben reinklicken) und alles mit Strg-Shift-V positionsgenau einfügen. Die EV-Zelle wird danach gelöscht.

2. Für diejenigen , die eine EV-Morrowind.esm verfügbar haben gibt es nch einen einfacheren Trick: Lade die Mod im TESCS mit der EV-MW.esm. benenne alle die geänderten englischen Interior-Cells, für die es in deutsch entsprechende Zellen gibt einfach in die deutschen Namen um (genau auf Buchstaben und leerzeichen achten). Das dann gespeichert und mit der DV-morrowind.esm geladen hat den selben effect wie 1.

Dannach sollte die Tür/Amulett des neuen Dungeon wenigstens vorhanden sein und man kann die neuen cellen betreten

Stufe 3: Zusätzlich zur Stufe 2 gibt es NPC mit neuen Dialogen und neue Scripte.

Hier kann es Probleme geben (und die gibt es garantiert) wenn irgendwo ein Bezug auf einen EV-Cell - Namen erfolgt.

Diese müssen (im ersten Schritt) alle umbenannt werden.

Also alle Bezüge im Dialogsystem und in Scripten suchen und ändern

Um bestimmte Texte zu finden gibt es die Funktion -->Edit, ->Find text

Dort den englischen Cellnamen oder den Topic-Namen eingeben und es werden alle Einträge in Dialog und Script gefunden die zu ändern / anzupassen sind.

--> Sehr wichtig ist das Fahnden nach versteckten Posiitionierungsbefehlen in Scripten, weil durch solche Fehler Gegenstände verschwinden oder nicht auftauchen

Also mit Findtext-> Scripte : "Positioncell" nach solchen suchen und auf die deutschen Cell-Namen ändern.

Danach sollten große Teile des Spiels in englisch funktionieren.

Stufe 4: Übersetzten von Dialogsystemen

Beim Übersetzen von Dialogsystemen sollte man sich vorher über den Aufbau und Funktion des Dialogsystems informieren. Es ist sehr wichtig die Abarbeitungsreihenfolge, Bedingungen und Result beurteilen zu können.

Im Prinzip funktioniert der Dialog jedes NPC nach folgenden Schema

Bei jedem Aktivieren des NPC -> Greeting je nach Bedingung (Such-Reihenfolge 1-->9) -> Textanzeige (mit Topics)-> RESULT: Journaleintrag oder neuer Topic wird freigeschaltet oder anderes

Anklicken eines TOPIC -> Bedingung ? -> Textanzeige (mit Topics)-> RESULT -> und wieder ein Topic oder anderes
usw.

Das Problem beim Übertragen des Dialogsystems sind die Texte, die Topics in englisch enthalten.

Wenn diese Texte übersetzt werden, werden auch die Topics mit übersetzt und der Ablaufzusammenhang ist solange gestört, bis neue Topics in deutsch (sinngleich) mit dem Textinhalt der englischen Topics bereitgestellt und aktiviert werden (addtopic).

Beispiel:

Steht im Result des greetings: addtopic "Longblade" und ihr übersetzt den Text wo eigentlich "Longblade" drin vor kommt und macht daraus "Langschwert" ist der Zusammenhang solange zerstört, bis Ihr auch einen Topic "Langschwert" anlegt (oder ihn umbenennt) mit den selben Bedingungen, Result und Inhalt /text von "Longblade"
und alle darin enthaltenen topics natürlich dann auch. usw.usf.

Ergänzung:

Ich habe die Erfahrung gemacht, das es sehr schwierig ist den Überblick zu behalten. Daher gehe ich jetzt wie folgt vor:

Zuerst den Anfang des Quests suchen. Der ist meistens in den greetings zu finden , aber oft auch in Topics wie "Last Rumors" "My trade" oder ähnlichen allgemeinen Topics.
Wenn man den Anfang hat (und eigentlich immer bevor man was ändert auf "Update hyperlink" drücken um die Verknüpfungen zu sehen. Den englischen Topic notieren.

Nach dem Übersetzen die Topics sofort umbenennen
und dann den gesamten Baum des Quest einschließlich Journale, Scripts logisch nach vollziehen und übersetzen.

Mit Findtext kann man die Logic auch über die
Suchfunktion nach Journal xxxxx nachvollziehen.

Erst wenn ein Quest vollständig übersetzt ist kommt der nächste dran.

Desweiteren können Fehlermeldungen entstehen, wenn EV-Greetings in Ihrem Result einen Bezug auf einen "TOPIC" haben und dieser ein Topic des original EV-Spiels ist, der aus der MW.esm geladen wird. Dieser englische Topic ist in der DV dann nicht vorhanden und es kommt zu Fehlern im Spielablauf.

Für den Fall der Verweise auf Original-Topics vorhanden sind, müssen die Verweise auf die original deutschen Topics umgeändert werden. (addtopic-Befehle ändern)

So, wenn das alles vollständig wieder verknotet ist, sollte das Spiel wieder mit Dialogen und Quests funktionieren.

Wenn man alles übersetzt, wird dann fast eine deutsche Mod daraus.

Stufe- Nebenbei: um das "fast" weg zu kriegen, braucht man nur noch alle Objektnamen zu übersetzten. Das ist von allem anderen unabhängig durchführbar.

- kein Anspruch auf Vollständigkeit -
nur eine schnell zusammengewürfelte Abfolge --meiner -- "TO-DO"- Reihenfolge.

Ich hoffe es hilft Euch weiter.

hochbass

Troubleshooting

12.: Häufig auftretende Probleme

Es gibt Probleme, die immer und immer wieder auftreten. Genau diese möchte ich an dieser Stelle erläutern.

Erst hier nimmt dieses Tut die Form eines FAQ's an, aber das soll uns nicht weiter stören.

Q.: Nachdem ich ein Plugin verändert und ein Savegame geladen habe sind alle Objekte doppelt vorhanden.

A.: Das liegt daran, das die Änderungen eines Plugins auch im Savegame gespeichert werden. Um diesen Bug zu umgehen, muss das Savegame einmal mit deaktiviertem Plugin geladen werden, wobei Fehlermeldungen auftreten werden. Dann muss man es erneut speichern und kann nun wieder das Plugin aktivieren.

Aber Achtung! Wenn es sich beispielsweise um ein Hausplugin handelt gehen alle Items in diesem Haus verloren! diese sollte man vorher in Sicherheit bringen.

Man kann diesen Fehler von vornherein ausschließen indem man während der Testphase eines Plugins nicht speichert.

Q.: Ich besitze die Low-Budget-Version, habe keine Editor-CD und demzufolge keine Meshes, Texturen, etc. Gibt es eine Möglichkeit wie ich sie trotzdem bekomme?

A.: Ja, das Spiel holt sich diese Sachen aus einem .bsa Archiv. auf der Website von Argent (<http://members.optusnet.com.au/argent2/>) gibt es ein Tool das diese komplett entpacken kann.

Q.: Ich habe einen Dialog gemacht aber es erscheint ein ganz anderer Beitrag als der den ich haben will.

A.: Das liegt daran das Morrowind die Dialoge von oben auch unten abklappert. Dabei wird das erste genommen, das alle Bedingungen erfüllt.

Daher sollten die komplexeren Themen immer weiter nach oben verschoben werden (mit der linken und der rechten Cursortaste).

Q.: Ich verschiebe mehrere Objekte am Grid aber die anderen sind manchmal nichtmehr am Grid ausgerichtet.

A.: Wenn man mehrere Objekte verschiebt wird immer das erste am Raster ausgerichtet und der Abstand zu den Übrigen beibehalten - unter Umständen muss von Hand nachgearbeitet werden.

Q.: Das Renderfenster bleibt grau.

A.: Desktop Farbtiefe auf 32Bit stellen

Q.: Mir gefällt das xyz aus dem Plugin zyx sehr gut, darf ich es in meinem Plugin verwenden?

A.: Generell sollte man immer nachfragen, sofern in der Readme des Plugins nichts anderes steht. Selbstverständlich sollte man den Entwickler dann auch in der Readme vermerken.

Q.: Ich habe eine EV Mod die ich gerne auf meiner DV spielen möchte, ist das möglich?

A.: Das Problem liegt darin das dir Zellen anders heißen und demnach nicht gefunden werden. Wer will kann sich natürlich dransetzen und das Plugin für die DV fit machen (-> S.56)

Q.: Wenn ich die Morrowind, Tribunal und Bloodmoon.esm gleichzeiti lade gibt es einen Haufen von Fehlermeldungen!

A.: Das ist vollkommen normal. Man kann sich das Leben erleichtern, indem man in der Morrowind.ini folgenden Eintrag hinzufügt:

AllowYesToAll =1

Q.: Wie mache ich sprechende Kreaturen?

A.: Eigentlich nicht schwierig.

Erstmal benötigt man eine Kreatur mit eigener ID, dort stellt man dann unter *AI Attack* auf einen Wert von oder unter 30. Nun muss man dieser Kreatur einfach ein eigenes Greeting verpassen -fertig!

Danksagung

Wenngleich ich es geschrieben habe, habt ihr dieses Tutorial nicht allein mir zu verdanken. Deshalb möchte ich an dieser Stelle den Leuten danken die mir bei der Fertigstellung geholfen haben sowie einige Teile beitragen und in naher Zukunft beitragen werden.

Mein Dank geht an:

Die User des Morrowindforum.de

Ohne sie wär ich niemals auf die Idee gekommen sowas durchzuziehen.

Die PlanetMorrowindModders

Bei denen ich die Möglichkeit hatte, meine Fähigkeiten weiter zu verbessern.

Bethesda

Für das Spiel an sich und natürlich den Editor.

Die gesamte restliche Modder-Community

Bei der ich soviel lernen konnte und die durch ihre Tools und Mods alles am laufen halten.

Mein besonderer Dank geht an:

Razorblade

Für ständige psychische Unterstützung und treue Freundschaft.

Garak

Für viele informative Gespräche und Belehrungen.

Iladrion

Dafür das er mir in naher Zukunft hoffentlich sein TESAME Tut zur Verfügung stellt ;-).

Hochbass

Für das Tutorial zur Umsetzung von EV Mod's.

XenArien

Für das Minitut mit den Gallerien.

lord.of.amaya

Für das Tut über sauber modden (ich habs ein wenig verbessert)

nullsoft

Ohne dessen Winamp ich nicht durchhalten würde.

Verbesserungsvorschläge sind in höchstem Maße willkommen

mfg

madmaxx