

# Gothic

## *A-2: Items, Objects & Events*

Letzte Änderung: 06.06.99 22:44

Druckdatum: 03.11.98 08:58

Autor: Mario Röske

© 1997,98, 99 Piranha Bytes Software GmbH

### Inhaltsverzeichnis

<b>1.</b>	<b>GEGENSTÄNDE</b>	<b>2</b>
1.1	FLAGS	2
1.2	SCHABLONE	5
1.3	STABILITÄT	5
1.4	GEWICHT	6
1.5	WERT	6
1.6	BENÖTIGTE ATTRIBUTE	6
1.7	BESITZ-KENNUNG	6
1.8	INSTANZIERUNG	6
1.9	INTERAKTION MIT GEGENSTÄNDEN	7
1.10	REFERENZ	11
<b>2.</b>	<b>TRAGBARE OBJEKTE</b>	<b>15</b>
2.1	STRUKTUR	15
2.2	INTERAKTION	17
<b>3.</b>	<b>GROSSE OBJEKTE</b>	<b>18</b>
3.1	STRUKTUR	18
3.2	INTERAKTION	20
<b>4.</b>	<b>STATIONÄRE OBJEKTE</b>	<b>21</b>
4.1	STRUKTUR	21
4.2	INTERAKTION	23
<b>5.</b>	<b>PHYSIK DER OBJEKTE</b>	<b>23</b>
<b>6.</b>	<b>EVENTS</b>	<b>24</b>
6.1	STRUKTUR	24
6.2	EVENTSKRIPT	27
6.3	GLOBALE EVENTS	27

# 1. Gegenstände

Ein Gegenstand wird durch folgende Struktur beschrieben:

- Flags
- Schablone
- Stabilität
- Gewicht
- Wert
- Benötigte Attribute
- Besitz-Kennung

## 1.1 Flags

Jeder Typ von Gegenstand kann mit einem oder mehreren Flags versehen werden. Diese Flags entscheiden darüber, was der SC mit dem Gegenstand anstellen kann und welche weiteren Daten darüber abgespeichert werden müssen. Folgende Flags sind vorgesehen:

Flag	Abk.	Beispiele
nahkampffähig	nf	Schwert, Axt, Dolch
fernkampffähig	ff	Bogen, Armbrust
verschießbar	vs	Pfeile, Bolzen
rüstbar	rb	Rüstungsteile
verzehrbar	vz	Essen, Tränke
lesbar	lb	Schriftrolle, Karte, Buch
schlösseröffnend	sö	Schlüssel, Dietrich
lichtspendend	ls	Fackel
zahlungsfähig	zf	Zahlungsmittel
- (sonstiges)	-	Symbol, Statuette, Feder, Scherbe, Kristall, Wappen
brennbar	bb	(passives Flag)
schwimmfähig	sf	(passives Flag)

Sind für einen Gegenstand mehrere Flags gesetzt, so muß eines davon als Hauptflag gekennzeichnet werden. Dieses Hauptflag ist nichts anderes als die Kategorie des Gegenstandes, die auch durch die Einteilung der Gegenstände (siehe I.1.10 „Referenz“) vorgenommen wurde.

Für jedes gesetzte Flag muß der Gegenstand über entsprechende zusätzliche Eigenschaften verfügen, die nur für die Verwendung mit diesem Flag nötig sind:

Flag	zusätzliche Eigenschaften
nahkampffähig	Schaden, Angriffsbonus, Parrierbonus, Anzahl Hände
fernkampffähig	Schaden, Angriffsbonus, Reichweite, Munition, Anzahl Hände, Nachladezeit
verschießbar	-
rüstbar	Schutz, Stelle, Tarnwirkung
fallentauglich	Schaden, Auffälligkeit
verzehrbar	Heilwirkung
lesbar	Text
schlösseröffnend	Schloss, Öffnungsbonus anderer Schlösser
lichtspendend	Helligkeit, Leuchtmodus
zahlungsfähig	Wert

Im folgenden werden die zusätzlichen Eigenschaften näher beschrieben:

### 1.1.1 Schaden

Waffen und Fallen fügen im Falle des Treffens bzw. Auslösens dem Opfer Schaden in Form von Lebenspunktabzug zu. Für Gegenstände existieren verschiedene Arten des Schadens:

Art des Schadens	Zeichen
Waffenschaden	
Feuerschaden	~
Magieschaden	*

Um die Art des Schadens in den untenstehenden Tabellen anzugeben ist das Zeichen hinter der Schadenshöhe selbst angegeben.

### 1.1.2 Munition

Für alle Gegenstände mit dem gesetzten Flag fernkampffähig muß die benötigte Art von Munition angegeben werden. Diese kann sein

Munitionsart	Abk	Anmerkungen
keine	-	Fernkampfwaffe wird selbst geworfen
Pfeile	Pfl	alle Arten von Bögen
Bolzen	Blz	alle Arten von Armbrüsten

### 1.1.3 Schutz

Fügt eine Waffe Schaden zu, so wird ein Teil dieser Schadenspunkte von den Rüstungspunkten absorbiert. Alle am Körper getragenen Rüstungsteile addieren ihre Schutzpunkte zu einem gemeinsamen Wert, der dann von den Schadenspunkten abgezogen wird.

### 1.1.4 Verkleidung

Alle Gegenstände die angezogen/getragen werden können (Rüstungen, Helme können den SC auf den ersten Blick als Angehöriger einer anderen Bande erscheinen lassen. Der „bemalte Lederwams der Alte-Lager-Torwachen“ oder aber der geschlossene Helm der Leibgarde des obersten Gurus können den Effekt haben, daß sich der Spieler verkleidet an Orte schleichen kann, die er sonst nicht erreichen würde.

Ein Rüstungsteil muß also einen Verkleidungs-Parameter haben, der entweder auf „keine Verkleidung“ oder aber auf dem Namen einer Bande steht.

### 1.1.5 Körperstelle

Obwohl der Schutz aller Rüstungsteile einfach aufaddiert wird, so können diese Gegenstände doch nur an bestimmten Stellen getragen werden:

Körperstelle	Anmerkung
Kopf	für Helme
Körper	für die eigentliche Rüstung

### 1.1.6 Heilwirkung

Nahrungsmittel verursachen beim Verzehr eine leichte Heilwirkung. Eine feste Zahl an LP-Symbolen wird zu den LP des SCs addiert.

### 1.1.7 Text

Gegenstände mit dem Flag lesbar müssen den entsprechenden Text enthalten. Dieser Text kann allerdings erst bei der Erzeugung eines tatsächlichen Gegenstandes (siehe 1.8. „Instanzierung“) dazugefügt werden. Um interessantere Aufträge gestalten zu können, sollten nicht nur Texte sondern auch Zeichnungen in Form von Karten möglich sein.

### 1.1.8 Schloß

Gegenstände, die das Flag *schlösseröffnend* gesetzt haben benötigen als Angabe die ID des passenden Schlosses. Dieses Schloß wird bei Benutzung dieses Gegenstandes automatisch geöffnet bzw. geschlossen. Auch dieser Wert kann erst bei der Instanzierung eines tatsächlichen Gegenstandes gesetzt werden, es sei denn es soll KEIN bestimmtes Schloß geöffnet werden (z.B. Dietrich).

### 1.1.9 Lichtparameter

Sämtliche notwendigen Lichtparameter hängen stark von der 3D-Engine ab und werden in [Appendix\Ac\\_Technik.doc - Licht](#) beschrieben.

### 1.1.10 Brenndauer

Lichtspendende Objekte haben eine begrenzte Leuchtdauer. Die Brenndauer ist ein Zähler, der festlegt, wie lange eine Fackel oder ein Leuchtstein brennt. Da Fackeln die wohl häufigsten lichtspendenden Gegenstände sein werden, möchte ich hier auf einige Besonderheiten der Fackeln stürzen.

- Fackeln sollen auf dem Boden liegend schneller verbrennen, als in der Hand gehalten.
- Geht die Brenndauer einer Fackel zur Neige, soll sie nicht einfach mit einem „Plöpp“ verlöschen sondern nach ca. 90% Brenndauer, kontinuierlich schwächer werden, anfangen wilder zu flackern und dann schließlich verlöschen (das ist wirklich cool, denn so ein richtiges Fackel-Flair gabs bisher noch nirgendwo)

Brandpfeile entzünden sich magisch. Brandpfeile, die auf etwas anderes als [Untote] treffen gehen sofort aus (→Rauchpartikel!)

#### 1.1.1.1 Ewiges Licht

Zusätzlich zur Brenndauer muß es ein Flag geben, mit dem das Zählen des Counters gestoppt werden kann. Anwendungsfälle:

- Fackeln, die in Wandhalterungen ewig brennen und beim Herausnehmen den Counter starten (Wichtig für Dungeondesign).
- Magische Leuchtsteine, die immer brennen.

### 1.1.11 Wert

Da jeder Gegenstand im Programm einen Wert besitzt, muß es auch etwas geben, mit dem für diesen Wert bezahlt wird. Die Zahlungsmittel stellen einen ganzzahligen Wert dar, der die Grundlage des Währungssystems bildet.

## 1.2 Schablone

Die Standardwerte aller Gegenstände sind, unabhängig von ihren Flags, in Schablonen festgehalten. Aus einer solchen Schablone wird dann im Spiel ein tatsächlicher Gegenstand generiert. Dies kann mit der objektorientierten Programmierung verglichen werden. Dort gibt es Klassen (die Schablonen unserer Gegenstände), die das Verhalten der Objekte bestimmen und Instanzen (bei uns die tatsächlichen Gegenstände) die aus den Klassen generiert werden.

Ein Beispiel. Die Gegenstandsschablone *Langschwert* bestimmt den angerichteten Schaden, die Strukturpunkte sowie das Gewicht. Wird ein Langschwert im Spiel benötigt, so wird daraus ein tatsächlicher Gegenstand generiert, der über die Werte Schaden, Strukturpunkte und Gewicht hinaus einen individuellen Namen (z.B.: „das rostige Langschwert“) erhält um diesen Gegenstand von anderen Gegenständen gleicher Schablone (genauer gesagt andere *Langschwerter*) zu unterscheiden.

## 1.3 Stabilität

- Waffen und Rüstungen gehen nie kaputt – Kein Schmied repariert Items (Schmiede könnten evtl. neue Sachen aus gefundenen Materialien herstellen)
- Objekte haben Empfindlichkeiten gegen verschiedene Schadenskategorien.
- Spitzhacke zerschlägt Stein. Stein muß vom SC gezielt zerstörbar sein, da versteinerte Wesen und falsch hingerollte Findlinge im Weg sein könnten
- Erzeugen neuer Objekte – Findling wird zu Felsen wird zu Steinen
- Verstreuung des Inhaltes einer Truhe, wenn diese zerstört wird

## 1.4 Gewicht

Es gibt zwar keine Gewichtsbeschränkung dessen, was der SC mit sich herumtragen kann (wird durch eine maximale Anzahl an Slots gelöst), allerdings ist das Gesamtgewicht trotzdem weiterhin von Bedeutung:

- Zusammenbrechen einer Brücke unter zuviel Gewicht
- Auslösen von Events durch Belasten einer Bodenplatte mit einem Minimalgewicht

Die Gewichtsdaten aller Gegenstandsschablonen sind in der Tabelle in [Appendix\Ad\\_Items.doc](#) aufgeführt.

## 1.5 Wert

Dies ist der Wert in Erznuggets (der Währung), die ein NSC für einen solchen Gegenstand haben will, wenn er ihn verkauft.

Die Werte für alle Gegenstandsschablonen sind in der Tabelle in [Appendix\Ad\\_Items.doc](#) aufgeführt.

## 1.6 Benötigte Attribute

Jedem Gegenstand können benötigte Attribute zugeordnet werden um es zu benutzen. So kann z.B. für ein Zweihänderschwert eingestellt werden, daß es erst ab Stärke 5 effektiv benutzt werden kann.

## 1.7 Besitz-Kennung

Sowohl Gegenstände als auch alle größeren Objekte, die weiter unten in diesem Kapitel beschrieben sind, können (müssen aber nicht) mit einer Besitz-Kennung versehen werden. Interagiert ein SC nun auf irgendeineweise mit diesem Gegenstand, so kann eine „Besitzer-NSC“ in der Nähe darauf reagieren.

Ein Besitz-Flag kann folgenden Infos beinhalten:

- Objekt gehört niemandem (keine Kennung)
- Objekt gehört einer bestimmten Bande
- Objekt gehört einem bestimmten Einzel-NPC

## 1.8 Instanziierung

Im Gegensatz zur Schablone, die nur die Defaultwerte eines zu erstellenden Gegenstandes darstellt, will ich den tatsächlichen Gegenstand selbst „Instanz“ (einer Schablone) nennen. Dies lehnt sich an die Nomenklatur von C++ an, wo es Klassen (unsere Gegenstandsschablonen) und Instanzen gibt.

### 1.1.12 Wann wird instanziiert ?

Jeder Auftrag beinhaltet zu instanzierende Gegenstände. Bei der Auftragsgenerierung (siehe Kapitel X „Auftragssystem“) wird also die Spielkarte mit tatsächlichen Gegenständen, den Instanzen „bevölkert“. Diese Instanzen werden in die Gegenstandsliste der Spielwelt aufgenommen.

### 1.1.13 Ablauf

Im Folgenden soll erläutert werden, wie aus einer Schablone eine Instanz gemacht wird.

#### 1.1.1.2 allgemeine Werte

Die allgemeinen Werte

- Flags
- Stabilitätspunkte
- Gewicht
- Wert

die für jede Schablone definiert sind, werden übernommen.

#### 1.1.1.3 zusätzliche Eigenschaften

Die in 1.1.1ff beschriebenen Eigenschaften werden dann in die Instanz übernommen, wenn die dazugehörigen Flags gesetzt sind.

Die beiden Parameter **Schloß** und **Text** nehmen eine Sonderrolle ein. Sie können nicht in der Schablone vorgegeben werden, da sie von Instanz zu Instanz unterschiedlich sein müssen. Vielmehr muß das passende Schloß zu einem Schlüssel sowie der enthaltene Text bzw. die Zeichnung eines Schriftstückes im Auftrag vorgegeben werden. Mehr dazu im Kapitel X „Auftragskonzept“.

#### 1.1.1.4 Magie

Der besondere Reiz vieler Gegenstände wird darin liegen, daß sie auf magische Art und Weise „veredelt“ wurden. Dazu kann jede beliebige Schablone bei Bedarf mit einem „Magie-Header“ versehen werden, der festlegt, wie die Magie des Gegenstandes funktioniert. Näheres dazu im Kapitel VII „Magiesystem“.

## 1.9 Interaktion mit Gegenständen

### 1.1.14 Werfen

Das Werfen von Gegenständen kann im Spiel vielfach angewandt werden:

- Werfen um zu erkunden: Fackel in Abgrund werfen um zu sehen, was da unten ist / Fackel über Schlucht werfen um zu sehen, ob man auf die andere Seite springen kann
- Werfen um wegzulocken: Stein werfen, um Wache abzulenken / Fleisch werfen, um Monster da hin zu locken / Meatbug werfen, um Minecrawler anzulocken / Gobboleiche in See werfen, um Snapper anzulocken / Besonders viel Krach könnte es machen, etwas auf einen Gong zu werfen oder eine Vase mit einem Steinwurf zu zerstören.
- Werfen um zu erreichen: Stein auf Knopf werfen, den man nicht erreichen kann / Stein auf Schatulle werfen, die oben auf einem Balken steht → Schatulle wird getroffen und fällt zu Boden (Schlüssel drin)
- Werfen, um zu bekämpfen: In der Schlucht sind Monster. SC steht oben und wirft Felsen hinab / Orks werfen Felsen im Fernkampf. Fackeln auf feuerempfindliche Untote werfen.

### 1.1.1.5 Vorgang

Der SC kann nur folgende Objekte werfen:

- kleine, baseballgroße Steine (werden durch „Benutzen“ von Steinhaufen-MOBs (→ Anim Aufheben) automatisch in die Hand des SCs generiert)
- brennende Fackeln (werden durch equippen einer neuen Fackel im Inventory oder aber durch Aufheben einer bereits brennenden Fackel in die linke SC-Hand gebracht und können mit ACTION + → in die rechte Wurfhand übertragen werden)
- tragbare Objekte (Kisten, Truhen, Felsbrocken)
- und Gobbos, falls man zufällig gerade ein Troll ist ;-)

Eine generelle Regel lautet:

- alles was wichtig ist landet beim Aufheben direkt im Inventory (Waffen, Rüstung, Schlüssel, Brief, etc.)
- alles was unwichtig (Dekoration, Stein, Blechnapf) ist, kann man in die Hand nehmen und

werfen.

Daher kann der SC auch nur diese Objekte in die Hand nehmen. Alle anderen Gegenstände wandern beim Aufheben sofort ins Inventory. Zieht der SC mit solchen Dingen in der Hand Waffen (oder Magie), schmeißt er das in der Hand befindliche Objekt vorher automatisch zur Seite und führt dann erst die entsprechende Aktion aus.

Geworfen werden kann nur aus dem Stand heraus; man bleibt stehen, drückt die ACTION-Taste, hält sie gedrückt, und mit der  $\uparrow$ -Taste wirft man maximal weit (Weite siehe unten).

#### 1.1.1.6 Ziel-Cursor

Um all Werfen-Anwendungen einfach zu steuern, bedarf es eines intelligenten Werfen-Fokus. Sobald ein Gegenstand in der rechten Hand bzw. ein trObj in beiden Händen der eigenen Spielfigur ist, und ACTION gedrückt gehalten wird, geht die Figur optisch in Wurfbereitschaft und ein Ziel-Cursor erscheint.

Dieser Ziel-Cursor muß folgende Eigenschaften haben:

- Mögliches Ziel:
  - Beliebige Punkte im Level
  - Figuren (aSCs, NSCs, Monster)
  - Objekte (Gegenstände, trObj, grObj, stObj)
- 3D-Cursor, das heißt der Cursor darf nicht einfach nur als Fadenkreuz auf den Screen projiziert werden, sondern muß einen Punkt im 3D-Raum kenntlich machen. Dies könnte man bei beliebigen Punkten im Level durch ein transparentes Ziel-VOB, bei Figuren/Objekten durch Highlighting/Umrahmung oder generell einfach durch Zentrieren des ViewPorts auf das Ziel realisieren.
- Zielerfassung:
  - Sobald die Wurfbereitschaft hergestellt ist (Gegenstand in Hand und ACTION gedrückt) wird die **maximale Wurfreichweite** ermittelt.
  - Dann wird ermittelt, ob sich eine **Figur** bzw. ein **Objekt** innerhalb der Wurfreichweite sowie innerhalb eines Sichtkegels der eigenen Figur (Blickrichtung  $+20^\circ$  (editierbar)) befindet.
  - Befinden sich ein oder mehrere Ziele im Wurfkegel, wird das am nächsten gelegene Ziel mit dem Cursor erfaßt.
  - Befinden sich keine Ziele im Wurfkegel, wird der Cursor auf einen **freien Punkt** im Level gesetzt. Dieser Punkt befindet sich auf der maximalen Wurfreichweite und in Blickrichtung des SCs.
  - Mindestwurfweite: Der Werfenfokus sollte nicht auf Objekte/NPCs reagieren, die zu nahe am Werfer stehen.
  - **Umschauen:** Wird zusätzlich die SCAN-Taste gedrückt, kann sich der SC mit den  $\leftarrow \uparrow \downarrow \rightarrow$ -Tasten (mit Wurfobjekt in der Hand) nach links/oben/unten/rechts umsehen und so auch Ziele anvisieren, die vom Autoaim der Figur (z.B. weit unten in einer Schlucht oder auf einem hohen Absatz) sonst nicht "gelocked" werden.

#### 1.1.1.7 Visualisierung

Alle Gegenstände werden unabhängig von der Wurfweite einhändig mit EINER Animation geworfen. Alle trObj zweihändig mit EINER anderen Animation.

#### 1.1.1.1 Wurfschaden

Alles, was geworfen wird macht Schaden entsprechend des Gewichtes. Objekte mit Nahkampfwaffen-Flag haben Schadensbonus (\*3). KEINE Speere! Alle Items mit einer Anim werfen. Schaden von kleinen Wurfgeschossen dient höchstens zum Ärgern (0 oder 1 Schadenspunkt)

#### 1.1.15 Aufnehmen von Gegenständen

Gegenstände können entweder offen herumliegen (Boden, Tische, Nischen) oder aber in einem Behälter sein (Truhe, Schrank, Leiche).

#### 1.1.1.8 Auf dem Boden

Gegenstände, die offen herumliegen müssen in den Fokus des Spielers geraten, um sie aufzunehmen. Ist dies der Fall, hängt es von der Art des Gegenstandes ab, was passiert:

Werfbare Items (z.B. kleine Steine und brennende Fackeln) werden mit ACTION + ↑ (Aufhebe-Anim) in die Hand genommen und können NICHT im Rucksack verstaut werden. Alle anderen Gegenstände (inklusive nicht brennender Fackeln) verschwinden nach der Aufhebe-Anim sofort im Rucksack.

#### 1.1.1.9 In Behältern

Gegenstände, die sich in Behältern „verstecken“, werden wie in [Appendix\Ab\\_UI.doc - OffeneBehälter](#) erklärt in das eigene Inventory befördert. Die Sonderstatus-Gegenstände *kleine Steine* und *brennende Fackeln* dürfen sich NIE in einem Behälter befinden, da sie sich immer nur in der Hand befinden dürfen.

#### 1.1.16 Fackeln

Wir haben es lange gedreht und gewendet aber es hilft alles nichts: Fackeln müssen eine Sonderrolle unter den Gegenständen einnehmen. Sie können weder als einfache Waffen zählen (dann kann man weder Leute anreden, noch Hebel ziehen, noch Kisten öffnen während man so ein Ding trägt) noch sind sie ein einfacher werfbarer Gegenstand (man muß sie zwar werfen, aber auch einfach nur nebenher tragen können). Kurzum: Hier kommt der Sonderfall Fackel:

#### 1.1.1.10 Fackelarten

Es gibt genau drei Arten von Fackeln.

- nagelneue, nicht brennende Fackeln (Inventory-Gegenstand)
- brennende Fackeln (Hand-Gegenstand)
- unbrauchbar, abgebrannte Fackeln (kein richtiger Gegenstand, sondern nichtaufhebbarer „Müll“)

Dazwischen gibt es keinerlei Zwischenstufen. Dummerweise muß im User-Interface jede Art von Fackel anders behandelt werden.

#### 1.1.1.11 Handhabung

Findet der SC eine *neue Fackel*, kann er sie, wie alle anderen Gegenstände, aufnehmen. Sie verschwindet dann direkt im Inventory und wird in den entsprechenden Multi-Slot zu den anderen *neuen Fackeln* gelegt. Der *neue-Fackel*-Zähler erhöht sich um eins.

Das gleiche Spiel läuft ab, wenn der SC *neue Fackeln* von einem NSC kauft.

Sollte der Spieler eine *neue Fackel* im Inventory equippen, wird sie von dort entfernt, angezündet und dem SC in die linke Hand gegeben (der SC könnte die dann schon *brennende Fackel* mit der linken Hand hinter seinem Rücken hervorziehen, da er ja durch den Inventorymodus ohnehin vor vorne zu sehen ist.). Ab jetzt ist sie eine *brennende Fackel* und kann **nie** wieder eine *neue Fackel* werden. Der SC kann auch eine bereits *brennende Fackel* (mit der linken Hand) aus einem Fackelhalter oder vom Boden nehmen. Diese bereits *brennende Fackel* wandert dann sofort in die linke Hand, da *brennenden Fackeln* ja keine Inventory-Gegenstände, sonder Hand-Gegenstände sind.

*Brennende Fackeln* sind also keine Inventory-Gegenstände sondern Hand-Gegenstände. Gut, das heißt, sie können nur in einer Hand gehalten und **nie** ins Inventory gepackt werden.

Trägt der SC eine *brennende Fackel* erst einmal in der linken Hand, bleibt der **User-Interface-Modus** im Normalmodus, und der SC kann alle Aktionen durchführen, die er auch ohne Fackel im Normalmodus könnte. Dies sind alle SC-Bewegungen ohne ACTION/MOVE-Tasten, alle Bewegungen mit MOVE-Taste und alle (friedlichen) Aktionen mit ACTION-Taste.

Sobald die **Leuchtdauer** der Fackel abgelaufen ist, wird sie zur *abgebrannten Fackel*. Solche Fackeln sind keine eigentlichen Gegenstände. Sie werden entweder als nichtaufhebbare „Müllobjekte“ in den Level gelegt (Decals?) oder besser verschwinden einfach in einer Partikelanimation (Realismus hin oder her, abgebrannte Fackeln will und braucht niemand).

#### 1.1.1.12 Werfen

Da man im Werfen keine der ACTION-Kommandos zur Verfügung hat, man die brennende Fackel aber auch Werfen können soll (in und über Abgründe, oder aber brennbaren Untoten an den Kopf :- ) ist es möglich mit ACTION + ←/→ die brennende Fackel zwischen der Ruhehand (links) und der Wurfhand (rechts, wie für alle anderen Wurf-Gegenstände auch) zu wechseln. Die SC-Figur führt dazu natürlich eine „Hand-Wechsle-Dich-Anim“ aus.

Sobald sich also die brennende Fackel in der rechten Hand befindet, wird das User-Interface auf **Werfen-Modus** geschaltet, und alles funktioniert wie beim Werfen aller anderen Wurf-Gegenstände auch. Siehe [Appendix\Ab\\_UI.doc - Werfenmodus](#)

## 1.2 Referenz

Alle Tabellen zu Items und Waffen sind in dem Doc-File [Appendix\Referenz\\_Items.doc](#)

[Nahkampfwaffen](#)

[Fernkampfwaffen](#)

[Munition](#)

[Rüstung & Schilde](#)

[Nahrungsmittel](#)

[Schriftwerk](#)

[Schlüssel](#)

[Lichtquellen](#)

[Zahlungsmittel](#)

[Sonstiges](#)[Tränke](#)[Magische Artefakte](#)

## 2. Tragbare Objekte

Nach den (Inventory-)Gegenständen sind dies die nächst größeren Objekte. Tragbare Objekte (Abkürzung: trObj) definieren sich dadurch, daß sie ALLE zu groß sind, um im Inventory verstaut werden zu können, aber klein genug, um sie in beiden (freien) Händen vor sich herzutragen. Dies muß dem Spieler schon von der Optik der Objekte ganz klar erkennbar sein.

### 1.10 Struktur

Ein tragbares Objekt wird (regeltechnisch) durch folgende Daten beschrieben:

- Typ
- Behälter?
  - Schloß ?
  - Falle ?
- Stabilitätspunkte
- Magie
- Besitz-Kennung

#### 1.1.17 Typ

Folgende Typen von trObj's sind vorgesehen:

VOB-Typ	Behälter ?	Schloß ?
Holzboxe	ja	ja
Schemel	nein	nein
kleines Faß	ja	nein
kleiner Felsbrocken	nein	nein
Gobbo-Leichen		

#### 1.1.18 Behälter

Einige Typen von tragbaren Objekten können auch als Behälter fungieren. Das heißt, sie können geöffnet und geschlossen werden und man kann Gegenstände darin ablegen bzw. von dort entnehmen (siehe [Appendix\Ab\\_UI.doc - OffeneBehälter](#)).

##### 1.1.1.13 Schlösser

Darüber hinaus kann bei einigen Behälter-trObj's auch ein Schloß angebracht sein, um den freien Zugang zum inneren des Behälters zu erschweren.

Ein Schloß kann geöffnet werden durch:

- Aufschließen mit dem richtigen Schlüssel
- Knacken mit einem Dietrich und dem Talent *Schlösser öffnen* (siehe **Fehler! Verweisquelle konnte nicht gefunden werden.**)
- Zerstören des trObj mit einer Waffe → Inhalt purzelt heraus
- Zerstören des trObj durch Werfen/Herunterwerfen → Inhalt purzelt heraus

#### 1.1.1.14 Fallen

Sobald ein trObj über ein Schloß verfügt, besteht auch die Möglichkeit, dieses Schloß mit einer Falle zu versehen.

*(Die einzelnen Objektfallen müssen noch festgelegt werden. Sie werden sich aber in jedem Fall aus grafischem Effekt und Schadensart/-höhe zusammensetzen.)*

#### 1.1.19 Stabilitätspunkte

Alle Objekte (also auch trObj) haben einen Stabilitätswert. Wirkt Schaden auf das Objekt ein, egal welcher Art, der größer als die Stabilität des Objektes ist, wird das Objekt sofort zerstört (VOB verschwindet hinter einem Partikeleffekt). Ist der Schadenswert dagegen kleiner als die Stabilität, so bleibt das Objekt unversehrt. Liegt es dabei auf dem Boden, springt/rutscht es in eine zufällige Richtung der Schadensquelle etwas davon.

#### 1.1.20 Magie

Auch trObjs können, wie schon Gegenstände mit „Magie-Headern“ versehen werden. Mehr dazu in **Fehler! Verweisquelle konnte nicht gefunden werden.** „Fehler! Verweisquelle konnte nicht gefunden werden.“.

#### 1.1.21 Besitz-Kennung

Wird genau wie bei Gegenständen gehandhabt (siehe 1.7 „Besitz-Kennung“).

#### 1.2.1 Visualisierung

Hier sind folgende Punkte wichtig:

- Die Größe und Wichtigkeit des Meshes muß den Spieler klar erkennen lassen, daß das Objekt zu groß ist, um es ins Inventory zu packen, aber auf der anderen Seite auch noch nicht so groß ist, daß es nur verschoben werden kann.
- die Greifweite des SCs ist 50 cm. Alle zweihändig tragbaren Gegenstände sollten ca. 50 cm breit sein

### 1.11 Interaktion

Mit trObj kann auf folgende Art und Weise interagiert werden:

Aktion	Erklärung
--------	-----------

aufnehmen	Hat der SC beide Hände frei (kein Kampfmodus) und steht er vor einem trObj, so nimmt er nach Drücken von ACTION + ↑ das trObj auf (Aufnehm-Anim) und hält es dann vor sich in den Händen
ablegen	Hält der SC ein trObj in Händen und drückt der Spieler ACTION + ↓, so legt der SC das Objekt an Ort und Stelle ab.
tragen	Hält der SC ein trObj in Händen, kann er sich nur noch eingeschränkt bewegen. Es ist nicht mehr möglich: Rennen, springen, klettern, schwimmen.
werfen	siehe 2.2.1 „Werfen“
verschieben durch Dagegenlaufen	Ein am Boden stehendes trObj kann durch einfaches Dagegenlaufen verschoben werden, wenn es weniger als <b>100 Steine</b> wiegt.. Dabei wird der SC langsamer. Ist das Objekt schwerer, rennt die Figur wie gegen eine Wand.
öffnen	Ist ein trObj zugleich ein Behälter (siehe unten), kann er geöffnet werden. Dabei wird eine eventuell vorhandene Falle ausgelöst, falls der SC nicht überdas Talent <i>Fallen entschärfen</i> verfügt .
schließen	Steht der SC vor einem Behälter-trObj, kann er durch bloßes Betätigen einer Taste diesen wieder schließen.
verschießen	Steht für ein Behälter-trObj mit Schloß der passende Schlüssel zur Verfügung, kann das trObj wieder verschlossen werden. Verschlossene Objekte werden NIE von NSCs geöffnet.
aufschließen	Ist ein Behälter-trObj verschlossen, so kann er durch das Anwenden das passenden Schlüssels aufgeschlossen werden.
knacken	Ist ein Behälter-trObj verschlossen, so kann er durch einen Dietrich und das Talent <i>Schlösser öffnen</i> geknackt werden.
angreifen/zerstören	Wird ein trObj mit einer Waffe oder Magie angegriffen, so nimmt sie ganz normal Schaden. Wird das trObj darüber hinaus zerstört, so bleiben Trümmer zurück. War das trObj auch noch ein Behälter, so liegen die enthaltenen Gegenstände nun verstreut auf dem Boden und können eingesammelt werden.
Gegenstand darin ablegen	Steht der SC vor einem geöffneten Behälter-trObj und legt er einen Gegenstand aus seinem Inventory ab, so landet dieser Gegenstand nicht auf dem Boden (wie sonst) sondern im Behälter-trObj (falls dort noch Platz ist).
Gegenstand daraus entnehmen	Steht der SC vor einem geöffneten Behälter-trObj und drückt er eine Taste so öffnet sich das Inventory des Behälters. Nun können die enthaltenen Gegenstände nach belieben ins eigene Inventory geschaufelt werden.

### 1.1.22 Werfen

Wenn man an einem Abgrund steht, kann man mittelgroße Gegenstände mit Autoaiming auf Ziele viel weiter unten werfen (Fokus reicht beim werfen zweihändig tragbarer Gegenstände wesentlich tiefer als normal)

Ansonsten läuft das Werfen genauso ab, wie das Werfen von Gegenständen (1.9.1). Der einzige Unterschied besteht darin, daß die Werfen-Animation eine andere ist (zweihändig).

Die Reichweite berechnet sich wie bei Gegenständen.

## 3. Große Objekte

Große Objekte (grObj) definieren sich dadurch, daß sie zu schwer/groß sind, um vom SC aufgehoben werden zu können. Dafür können sie langsam und mühsam am Boden geschoben/gezogen werden.

### 1.12 Struktur

Die Struktur ist ähnlich zu den tragbaren Objekten.

- Typ
- Schiebe-/Ziehparameter
- Behälter?
  - Schloß ?
  - Falle ?
- Stabilitätspunkte
- Magie
- Besitz-Kennung

### 1.1.23 Typ

Folgende Typen von grObj's sind vorgesehen:

VOB-Typ	Behälter ?	Schloß ?
Truhe	ja	ja
großes Faß	ja	-
Schrank	ja	ja
Tisch	-	-
umgefallener Baumstamm	-	-
Statue	-	-
Ballista	-	-
großer Felsbrocken	-	-
Große Portaltüren	-	ja
Geheimwände	-	ja

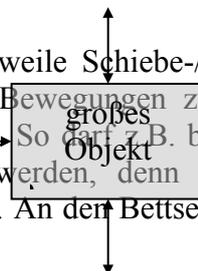
ANMERKUNG: Keine Betten! Betten immer Strohhaufen auf Boden oder niedriger, nicht verschiebbarer Sockel!

### 1.1.24 Schieben/ziehen

Das Schieben bzw. Ziehen von großen Objekten muß etwas näher beleuchtet werden. Für jedes grObj kann festgelegt werden, in welche Richtung es geschoben bzw. gezogen werden kann und in welche nicht. Dabei wird jedes grObj schematisch in einen bounding-box-artigen Quader eingepackt, der dann von oben betrachtet wird.

Für jeden dieser

Doppelpfeile muß also festgelegt werden, ob die jeweilige Schiebe-/Ziehrichtung erlaubt ist oder nicht. Prinzipiell könnte man natürlich jede dieser Bewegungen zulassen, doch aus grafischen Gründen müssen einige Richtungen verboten werden. So darf z.B. bei einem normalen Bett (sehr niedrig) nur am Kopf- und Fußende geschoben werden, denn dort befindet sich auch ein Ansatzpunkt für die Hände in der Schiebe-Animation. An den Bettseiten dagegen würde der SC in der Schiebe-Anim das Bett garnicht berühren.



#### 1.1.1.15 Vorgang

Der SC hat ein grObj im Fokus und kann nun ACTION drücken. Tut der Spieler dies, so geht seine Figur in die Schiebe-/Ziehgrundhaltung (s\_PStand). Hält der Spieler nun immer noch ACTION gedrückt und drückt dann zusätzlich ↑, beginnt sein SC, das grObj zu schieben. Zusätzliches Drücken von ←/→ zu ACTION + ↑ bewirkt ein Schieben um die Kurve (wie beim Gehen/Rennen halt auch).

Soll das grObj anstatt dessen gezogen werden, muß lediglich ACTION + ↓ gedrückt werden. Auch hier kann mit ←/→ um die Kurve gezogen werden.

#### 1.1.1.16 Verschieben und benötigte Stärke

Es wird einen digitalen Schwellwert geben. Hat der SC **weniger als 5 Pöppel Stärke**, kann er große Objekte nicht verschieben. Er lehnt sich dagegen, müht sich ab, aber das Objekt wackelt nicht die Bohne.

Verfügt der SC dagegen **über 5 Pöppel Stärke** oder mehr, kann mit einer angestregten Verschiebe-Anim das Objekt langsam bewegt werden. Die Geschwindigkeit ist wieder mal rein von der Anim abhängig. SC setzt immer in gleicher Höhe (1m) zum Schieben und Ziehen an. Seitliches Lenken möglich.

#### 1.1.1.17 Grenzen

Beim Leveldesign wird darauf geachtet, daß grObj auf ebenen Böden mit Umrandungen plaziert werden. Die Umrandung kann z.B. in Form einer „Teppichleiste“ bzw. einer Absenkung realisiert werden, aus welcher z.B. ein Schrank nicht herausgeschoben werden kann.

Wird nun ein grObj von einem Ebenen Boden auf eine nach oben führende Schräge bzw. sogar auf eine Wand (Schräge mit 90°) geschoben, wird die Bewegung abgebrochen und das grObj bleibt stehen.

Hier eine Seitenansicht:

#### 1.1.1.18 Abrutschen/Zerschellen

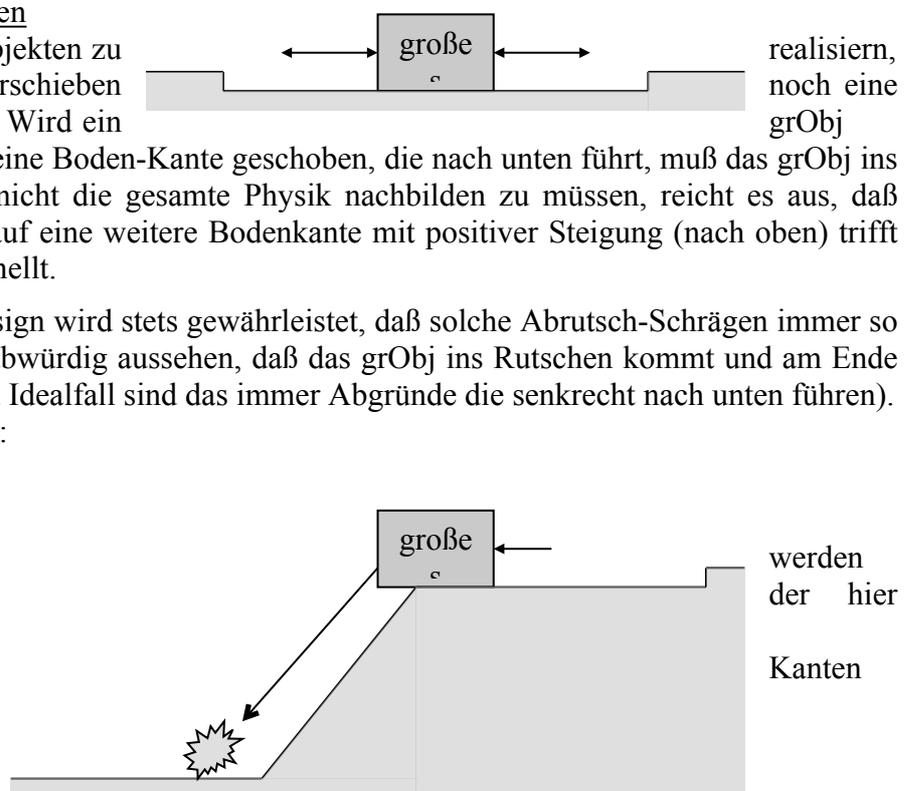
Um interessante Rästel mit Objekten zu realisieren, noch eine grObj sollte neben dem bloßen Verschieben weitere Spielart möglich sein. Wird ein von einem ebenen Boden auf eine Boden-Kante geschoben, die nach unten führt, muß das grObj ins Rutschen kommen. Um nun nicht die gesamte Physik nachbilden zu müssen, reicht es aus, daß sobald das rutschende grObj auf eine weitere Bodenkante mit positiver Steigung (nach oben) trifft umfällt und automatisch zerschellt.

Durch aufmerksames Leveldesign wird stets gewährleistet, daß solche Abrutsch-Schrägen immer so steil sind, daß sie optisch glaubwürdig aussehen, daß das grObj ins Rutschen kommt und am Ende der Rutschpartie zerschellt (im Idealfall sind das immer Abgründe die senkrecht nach unten führen). Hier wieder eine Seitenansicht:

Ebenfalls

im

Leveldesign berücksichtigt muß, daß die Rutschfläche in nicht dargestellten dritten Dimension **keine** weitere enthält.



#### 1.1.25 Behälter,

#### Stabilitätspunkte & Magie

Siehe 2. „Tragbare Objekte“.

### 1.1.26 Besitz-Kennung

Wird genau wie bei Gegenständen gehandhabt (siehe 1.7 „Besitz-Kennung“).

## 1.13 Interaktion

Mit grObj kann auf folgende Art und Weise interagiert werden:

Aktion	Erklärung
öffnen	Ist ein grObj zugleich ein Behälter, kann es geöffnet werden. Dabei wird eine eventuell vorhandene Falle ausgelöst wenn das Talent <i>Fallen entschärfen</i> nicht vorhanden ist.
schließen	Steht der SC vor einem Behälter-grObj, kann er durch bloßes Betätigen einer Taste diesen wieder schließen.
verschließen	Steht für ein Behälter-grObj mit Schloß der passende Schlüssel zur Verfügung, kann das grObj wieder verschlossen werden. Verschlossene Objekte werden NIE von NSCs geöffnet.
aufschließen	Ist ein Behälter-grObj verschlossen, so kann er durch das Anwenden des passenden Schlüssels aufgeschlossen werden.
knacken	Ist ein Behälter-grObj verschlossen, so kann er durch einen Dietrich und das Talent <i>Schlösser öffnen</i> geknackt werden.
angreifen/zerstören	Wird ein grObj mit einer Waffe oder Magie angegriffen, so nimmt sie ganz normal Schaden. Wird das grObj darüber hinaus zerstört, so bleiben Trümmer zurück. War das grObj auch noch ein Behälter, so liegen die enthaltenen Gegenstände nun verstreut auf dem Boden und können eingesammelt werden.
schieben	siehe oben
zeihen	siehe oben
Gegenstand darin ablegen	Steht der SC vor einem geöffneten Behälter-grObj und legt er einen Gegenstand aus seinem Inventory ab, so landet dieser Gegenstand nicht auf dem Boden (wie sonst) sondern im Behälter-grObj (falls dort noch Platz ist).
Gegenstand daraus entnehmen	Steht der SC vor einem geöffneten Behälter-grObj und drückt er eine Taste so öffnet sich das Inventory des Behälters. Nun können die enthaltenen Gegenstände nach belieben ins eigene Inventory geschaufelt werden.
Auslösen	z.B. mittels ACTION-Taste für Ballista

## 4. Stationäre Objekte

Stationäre Objekte (stObj) sind so groß/schwer (oder aber fest installiert), daß sie überhaupt nicht mehr vom Fleck bewegt werden können.

### 1.14 Struktur

Die Struktur ist ähnlich zu den anderen Objekt-Arten.

- Typ
- Status (z.B.: offen/geschlossen für Türen oder oben/unten für Schalter)
- Schloß ?
- Falle ?
- Stabilitätspunkte
- Magie
- Besitz-Kennung

### 1.1.27 Typ

Folgende Typen von stObj's sind vorgesehen. Eventuell können einige stObj zusammengefaßt werden, da sie sich nur im Material unterscheiden wie z.B. Holz- und Metalltür.

stObj-Typ	Behälter ?	Status ?	Schloß ?
Holztüre	-	ja	ja
Metalltüre	-	ja	ja
Steinblockpassage (Tür)	-	ja	-
Fallgitter	-	ja	-
Fallsteinblock	-	ja	-
Falltüre	-	ja	-
Geheimtüre (aufdrückbar, schwingt zur Seite, Stein)	-	ja	-
Geheimtüre (aufdrückbar, Drehtür, Stein)	-	ja	-
zerstörbare Holzwand	-	-	-
zerstörbare Steinwand	-	-	-
Altar	-	-	-
Sarkophag	ja	ja	-
Bodenschalter (muß belastet werden)	-	ja	-
Wandschalter (normler Hebel/Schalter)	-	ja	-
Dungeon-Falle	-	-	-
Drehspill (wie auf Schiffen der Ankerspill)	-	ja	-
Fackelhalter	ja	-	-
Halterung (Einsetzen von Rätselgegenständen)	ja	-	-
Ankerspill/"Conan"-Drehkreuz			
Zugkette			

### 1.1.28 Status

Neu bei stationären Objekten ist, daß sie einen „Status“ haben können. Dies ist ein digitaler Wert, der in den Skripten verankert sein muß. Meistens ist dies offen/geschlossen für Türen und ähnliches oder unbetätigt/betätigt für diverse Schalter.

Diese stObj müssen durch bestimmte Aktionen des SCs, meistens einfach nur Manipulation mittels der ACTION-Taste, ausgelöst werden. Ein Status ist durch eine andere SC-Aktion wieder umkehrbar.

Im Folgenden wird auf einige Supersonderspezialfälle von stationären Objekten eingegangen, um deren Funktionsweise genauer zu durchleuchten:

#### 1.1.1.19 Fallgitter

Fallgitter müssen mit einem Mechanismus (ein anderes stObj) wie z.B einem Hebel, einem Drehkreuz oder einer Kette geöffnet/geschlossen werden.

Dinge, die sich im Wege der „Fallgitter-Fall-Zu-Animation“ befinden, nehmen Schaden:

- SCs und NSC nehmen Schaden wie durch eine getroffene Zweihandschwert-Attacke.
- Gegenstände werden auf eine Seite gedrückt.
- trObj/grObj werden „attackiert“. Übersteigt der Schaden des Fallgitters, die Stabilität des Objektes wird es zerstört und das Gitter ist zu, wenn nicht, wird das Fallgitter blockiert, federt dreimal ab und fährt wieder in die Ausgangsposition zurück.

Durch geschlossene Fallgitter kann man hindurchschießen.

### 1.1.1.20 Türen einschlagen

Verschlossene Türen können mit Nahkampfwaffen „geöffnet“ werden. Übersteigt der Schaden der Waffenattacke die Stabilität der Tür, fliegt diese auf.

### 1.1.29 Schloß, Falle, Stabilität & Magie

Siehe 2. „Tragbare Objekte“.

### 1.1.30 Besitz-Kennung

Wird genau wie bei Gegenständen gehandhabt (siehe 1.7 „Besitz-Kennung“).

## 1.15 Interaktion

Mit stObj kann auf folgende Art und Weise interagiert werden:

Aktion	Erklärung
öffnen	Siehe oben, allerdings muß das stObj kein Behälter sein. Es genügt daß ein Status vorhanden ist
schließen	dito
verschließen	Siehe oben. Das stObj muß kein Behälter sein. Es genügt ein Schloss
aufschließen	dito
knacken	dito
angreifen/zerstören	siehe oben
Gegenstand darin ablegen	dito
Gegenstand daraus entnehmen	dito
Auslösen durch Manipulieren	z.B. ACTION-Taste bei Wandschalter
Auslösen durch Belasten	z.B. Bodenplatte
Auslösen durch Einsetzen eines Gegenstands	z.B. Fackelhalter

## 5. Physik der Objekte

Hier einige allgemeine Regeln zu Items, trObj, grObj, stObj, die im Folgenden einfach nur Objekt genannt werden:

- Jedes Objekt besitzt eine Bounding-Box, in Form eines kleinstmöglichen Quaders um das Objekt herum.
- Für Objekte kann die Schwerkraft ein/ausgeschaltet werden (per Skript).
- Liegen Objekte mit Schwerkraft auf einer Kante des Levels, so „purzeln“ sie über die Kante, falls der graphische Schwerpunkt der Objekt-Bounding-Box über die Kante ragt.
- Liegen Objekte auf anderen Objekten, so bleiben sie darauf liegen, wenn der Schwerpunkt des oberen Objektes auf der Bounding-Box des unteren liegt. Sonst rutscht/purzelt das obere Objekt runter.
- Werden Objekte über einen Knick im Boden geschoben, der weniger als 30° beträgt, so werden sie automatisch „angeliftet“ und die Schräge hinaufgeschoben.
- Objekte, die rutschen, fallen, geworfen werden verursachen beim Aufprall je nach Gewicht Schaden. Prallen sie auf Figuren oder andere Objekte, wird der Schaden wie Waffenschaden

behandelt.

## 2. Events

### 2.1 Struktur

Events reagieren auf bestimmte Spielsituationen. Sie bestehen aus:

- Freischaltung
- Auslöser
- Verzögerung
- Auswirkungen
- Wiederholung

#### 2.1.1 Freischaltung

Alle im Spiel vorkommenden Events müssen von Spielbeginn an in der Karte eingebaut sein. Allerdings soll es möglich sein, einige Events erst einmal inaktiv ruhen zu lassen und erst bei Eintreten einer besonderen Spielsituation freizuschalten also zu aktivieren.

So kann durch die Annahme eines Auftrages ein Event freigeschaltet werden, daß bei vorheriger Erfüllung der Event-Auslösekriterien nicht ausgelöst werden soll. Dies könnte z.B. eine Bodenplatte sein, die bei Betreten einen NSC-Geist erzeugt (der SC wurde von seinem Auftraggeber dorthin geschickt um einen Geist um Rat zu fragen). Wird die Bodenplatte betreten, bevor der Auftrag erteilt wurde, so passiert nichts, da das Event zu diesem Zeitpunkt nicht freigeschaltet war.

#### 2.1.2 Auslöser

Folgende Möglichkeiten existieren, ein Event auszulösen:

Auslöser	Paramter	Beispiele, die möglich sein sollten
Levelstart	Startingpoint (es wird mehrere Startingpoints pro Level geben)	<ul style="list-style-type: none"> <li>• alle Türen beginnen, bei erstmaligem Betreten des Levels (ein großes Haus) ständig zu klappern</li> </ul>
Kollidieren mit Raumsegmenten	(unsichtbares) Trigger-VOB	<ul style="list-style-type: none"> <li>• das Betreten einer Marmorplatte löst eine Pfeilfalle aus</li> <li>• das Betreten der Bodenplatte vor einem Altar läßt einen Geist erscheinen, und zwar nur für den SC, der Auftrag 22 angenommen hat</li> </ul>
Benutzung eines Gegenstandes	Item-Instanz	<ul style="list-style-type: none"> <li>• das Blasen in ein magisches Horn öffnet eine Tür</li> <li>• Das Lesen einer Schriftrolle in einem Altarraum, läßt Skelettons erscheinen</li> </ul>
Aufnehmen eines Gegenstandes	Item-Instanz	<ul style="list-style-type: none"> <li>• Das Wegnehmen eines Kelches von einem Altar schließt alle Türe zur Grabkammer</li> </ul>
Benutzung eines VOBs	VOB- Instanz	<ul style="list-style-type: none"> <li>• das Betätigen eines Schalters hebt eine Glasvitrine an</li> <li>• das Herunterdrücken des Armes einer Statue öffnet</li> </ul>

		eine Geheimtür
Benutzung eines VOBs mit einem Gegenstand	VOB- und Item-Instanz	<ul style="list-style-type: none"> <li>• ein Schlüssel öffnet eine verschlossene Tür</li> <li>• Das Einsetzen eines Edelsteins in eine Statue öffnet eine Geheimtür</li> </ul>
NSC stirbt	NSC-Slot	<ul style="list-style-type: none"> <li>• nachdem der Meisteralchemist getötet wird fegt ein gewaltiger Feuerzauber durch das Labor (armer Mörder...)</li> </ul>
automatisch in regelmäßigen Zeitabständen	Wartezeit	<ul style="list-style-type: none"> <li>• eine kleiner Vulkan spuckt alle 3 Sekunden einen Feuerball aus</li> </ul>
Aufruf durch vorangehendes Event		<ul style="list-style-type: none"> <li>• In einem langen Gang sind zehn Türen untergebracht die, sich nach schließen der ersten Türe im Abstand von je 1 Sekunde nachfolgend schließen.</li> </ul>

Alle Arten von Auslösern können auch kombiniert werden. So könnte z.B. das Blasen in das Horn (Benutzung eines Gegenstandes) nur auf einer bestimmten Bodenplatte (Kollidieren mit Raumsegmenten) die Türe öffnen.

### 2.1.3 Verzögerung

Nachdem die Bedingungen zum Auslösen eines Events erfüllt sind kann mit diesem Eventparameter die Auswirkung des Ereignisses selbst verzögert werden.

### 2.1.4 Auswirkungen

Das Auslösen jeden Events zieht (nach der Verzögerung) bestimmte Auswirkungen nach sich:

Auswirkung	Paramter	Beispiele
Item erzeugen	Item-Instanz, Koordinaten-VOB	<ul style="list-style-type: none"> <li>• Die Benutzung eines magischen Steins an einem Altar läßt ein mächtiges Schwert auf ihm erscheinen.</li> </ul>
Item löschen	Item-Instanz	<ul style="list-style-type: none"> <li>•</li> </ul>
Item bewegen/rotieren	Item-Instanz, Richtungsvektor, Geschwindigkeit	<ul style="list-style-type: none"> <li>• Die Annäherung an ein magisches Schwert auf dem Boden, läßt dieses „wegkrabbeln“.</li> </ul>
VOB erzeugen	VOB, Koordinaten-VOB	<ul style="list-style-type: none"> <li>• Das Entfernen eines Stützbalkens in den Minenstollen läßt die Stollendecke einstürzen (→ Geröll-VOBs werden im Gang erzeugt)</li> </ul>
VOB löschen	VOB	<ul style="list-style-type: none"> <li>• Das Umlegen eines Hebels läßt einen blockierenden Felsen verschwinden.</li> </ul>
VOB bewegen/rotieren	VOB, Richtungsvektor, Geschwindigkeit	<ul style="list-style-type: none"> <li>• Das Umlegen eines Hebels läßt einige Treppenstufen absenken.</li> <li>• Das Betreten einer Bodenplatte öffnet eine Falltüre.</li> </ul>
NSC erzeugen	NSC-Instanz, Koordinaten-VOB	<ul style="list-style-type: none"> <li>• Das entnehmen eines Edelsteines aus einer Götze läßt einen Zorngeist erscheinen</li> </ul>
NSC löschen	NSC-Instanz	<ul style="list-style-type: none"> <li>• Wird der Edelstein wieder eingesetzt verschwindet der Zorngeist wieder.</li> </ul>
Lichtquelle manipulieren	Leuchart	<ul style="list-style-type: none"> <li>• manipuliert vorhandene Lichtquellen (Leuchart ein/aus/toggle/swell)</li> </ul>
Teleport SC	Zielkoordinaten	<ul style="list-style-type: none"> <li>• z.B. für Teleportfallen</li> </ul>
SC-Attribut ändern	Attribut, Änderung	<ul style="list-style-type: none"> <li>• SC-Attribute siehe III.1.3.1</li> </ul>
SC-Status ändern	Status, Änderung	<ul style="list-style-type: none"> <li>• SC-Status siehe III.1.4</li> </ul>
Multimediaausgabe		<ul style="list-style-type: none"> <li>• Musik, SFX, Sprachsample ausgeben</li> </ul>
Karte wechseln	Neue Karte, Startkoordinaten	<ul style="list-style-type: none"> <li>• der Spieler fällt durch eine Falltüre in ein anderes Level</li> <li>• der Spieler läuft einen Gang entlang, der sich im nächsten Level fortsetzt</li> </ul>

Hintergrund wechseln		<ul style="list-style-type: none"> <li>• Sternenhimmel, Wolkendecke, Berge, Regen, Schnee, Gewitter, Donner, usw. können im Hintergrund eingeblendet werden</li> </ul>
Magie auslösen	Spruchart + Spruchparameter	<ul style="list-style-type: none"> <li>• Das Trinken aus einem Brunnen löst einen Heilspruch aus.</li> </ul>
NSC verläßt die Party	NSC- Instanz	<ul style="list-style-type: none"> <li>• Der Spieler will zusammen mit seinem NSC-Begleiter eine gefährliche Höhle betreten, doch vor dem Eingang sagt der NSC: „Ab hier mußt Du alleine gehen.“</li> </ul>
Attitüde eines NSCs ändern	NSC- Instanz, Änderung	<ul style="list-style-type: none"> <li>• Der SC betritt einen verbotenen Raum und zieht den Zorn des Hausherrn auf sich</li> </ul>
NSC beginnt Dialog	NSC-Instanz	<ul style="list-style-type: none"> <li>• Der SC tritt in eine Halle und wird von einer Wache angesprochen.</li> </ul>
Event freischalten/sperrern	Event-Nr.	<ul style="list-style-type: none"> <li>• schaltet ein anderes Event im Spiel frei oder sperrt es (VORSICHT: kein Auslösen!!!)</li> </ul>
Folge-Event	Event-Nr.	<ul style="list-style-type: none"> <li>• Eine erste Tür die sich schließt zieht das sekundenweise Schließen anderer Türen hinter sich.</li> </ul>
NSC spricht einen Satz außerhalb des Dialoges	Satz	<ul style="list-style-type: none"> <li>• eine Wache sagt „Geh weg von der Tür!“ sobald der SC sich der Türe nähert</li> </ul>

Ein einzelnes Event kann mehrere Auswirkungen gleichzeitig nach sich ziehen. Darüber hinaus bietet die Auswirkung „Folge-Event“ die Möglichkeit, interessante Verkettungen von Ereignissen herbeizuführen wie z.B. eine Reihe von Leuchterscheinungen (Licht manipulieren), die immer aufs neue verzögert erscheinen und so dem Spieler einen „Leuchtpfad“ weisen.

### 2.1.5 Wiederholung

Wurde nun ein Ereignis ausgelöst so muß festgelegt werden, ob bei erneutem Eintreten der Auslösebedingungen das selbe Ereignis wieder ausgelöst wird oder ob es sich dabei nur um eine einmalige Angelegenheit handeln soll. Folgende Arten von Wiederholung existieren:

Art der Wiederholung	Erklärung	Beispiel
Einmalig	das Ereignis kann nur ein einziges Mal ausgelöst werden	eine Statue wird durch das Einfügen eines Stabes in eine Wandöffnung zersört
begrenzter Zahl von „Ladungen“	das Ereignis kann X-mal ausgelöst werden, wobei X eine in der Eventbeschreibung vorgegebene Zahl ist	das Betreten einer Bodenplatte löst eine Pfeilfalle aus, die nach 12 Mal keine Pfeile mehr enthält
Wartezeit bis zum nächsten Auslösen	das Ereignis kann nach dem Auslösen erst nach einer bestimmten Wartezeit erneut ausgelöst werden; dieser Zyklus kann allerdings beliebig oft wiederholt werden	das Berühren eines großen hell leuchtenden Kristalles löst einen Zauberspruch aus der heilt; der Kristall muß sich nun aufladen (Licht wird heller und heller) und kann erst dann wieder seine Heilmagie freisetzen
„Ladungen“ + Wartezeit	wie oben, nur kann der Zyklus nicht beliebig oft wiederholt werden, sondern nur X-mal	der Heilkristall von oben verliert nach 6mal seine magische Wirkung und bleibt für immer dunkel
Unendlich	des Ereignis kann immer und immer wieder ausgelöst werden	das überfliegen eines bestimmten Feldes führt immer wieder zu einem Windstoß (Magie), welcher den SC auf das letzte Feld zurückschleudert

## 2.2 Eventskript

Auch hier kommt die Skriptsprache zur Anwendung, die in einem gesonderten Dokument definiert ist.

## 2.3 Globale Events

Um bereits besuchte Örtlichkeiten der Spielwelt, die nun erforscht und vielleicht „gesäubert“ sind, auch im späteren Spielverlauf immer wieder interessant zu gestalten, müssen spielweltglobale Ereignisse in regelmäßigen Abständen ausgelöst werden, die solche Kartenbereiche erneut mit NSCs und Gegenständen bevölkern.

Regeltechnisch wird dies mittels Events mit dem Auslöser „automatisch in regelmäßigen Zeitabständen“ realisiert. Da diese Art von Events gegenüber den normalen Events einige zusätzlichen Parameter benötigen, will ich hier gesondert auf sie eingehen.

Es muß für jedes Globale Event festgelegt werden:

- Freischaltung
- Auslöser („automatisch in regelmäßigen Zeitabständen“)
- Verzögerung (keine)
- Auswirkungen (variabel, normalerweise aber „NSC erzeugen“)
- Wiederholung (variabel, normalerweise aber „unendlich“)
- Text für News-Infopool
- Einspeisung in welche Lager?

### 2.3.1 Text für News-Infopool

Um nicht den Eindruck des plumpen „Monsternachwachsens“ zu erwecken müssen diese globalen Ereignisse natürlich storytechnisch sauber in die Spielwelt integriert werden. Das bedeutet, daß die Auslösung dieses Mechanismus in Form einer Information in den News-Infopool einiger Lager aufgenommen wird. Hier einige Beispiele:

<b>globaler Event</b>	<b>Gerücht</b>
Verstärkung des Söldnerlagers	„Ich habe gehört, daß der Söldnerhauptmann neulich wieder neue Rekruten anheuert hat.“
die alte Mine lebt	„Letzte Woche wurden wieder streunende Wolfsrudel in der alten Mine gesehen.“
Friedhof des Grauens	„Jüngst berichtete ein völlig Verstörter von neuen seltsamen Geistererscheinungen auf dem Friedhof.“

### 2.3.2 Einspeisung in Lager

In welches der fünf Lager diese News eingespeist wird, muß ebenfalls festgelegt werden (siehe **Fehler! Verweisquelle konnte nicht gefunden werden.** „Fehler! Verweisquelle konnte nicht gefunden werden.“).

