

Explosionen

by Alexander Pippan aka RnB *rrr*bbb* <alexander.pippan@rnbcreative.at>, 31/08/01

Vorwort

Manche kennen vielleicht den Command P_Explosion, der nicht in der MaxEd.ini definiert ist und daher nicht angewendet werden kann. Falsch gedacht, er wird angewendet nur nicht direkt im MaxEd. Mehr dazu im Laufe des Tutorials. Im Tutorial machen wir jetzt einen Kanister der Explodieren und Schaden verursachen soll. Also los gehts:

Einstellungen

Im Tutorial verwende ich wieder einen Raum (StartRoom), mit PlayerStart und einen einfachen Block (Kanister). PlayerStart und Kanister wieder zum StartRoom gruppieren.

Den Block macht ihr anfangs einmal dynamisch (F5+d). Vergewissert euch das Kollisionen an sind. Im weiteren Verlauf müssen jetzt einige Messages eingefügt werden. Fügt bei der **Startup**-Kategorie des Kanisters folgendes ein:

this->DO_SetHitpoints(20); Das gibt jetzt unserem Kanister 20 Hitpoints bis er stirbt.

In der **DO_OnDeath**-Kategorie, also wenn er stirbt folgende Commands eingeben, die Erklärungen folgen danach:

- **this->Do_hide(true);**
- **this->P_CreateProjectile(Damage_mine, 1);**
- **this->P_CreateProjectile(decals_explosion, 1);**
- **this->A_Play3DSound(dynamic, explosion_large, "");**
- **this->ps_starteffect(explosion_gasoline_huge, "");**

Jetzt ist man schon fertig, wer die Commands verstehen will sollte sich noch die Erklärungen ansehen:

- **this->DO_Hide(true);** Sollte jedem klar sein, versteckt den Kanister sobald die Hitpoints verbraucht sind
- **this->P_CreateProjectile(Damage_mine, 1);** Hier wird angegeben wie groß der Schadensradius der Explosion sein soll. Es kommt da auch der Command P_Explosion ins Spiel. Der Command wird in einer Textdatei im Verzeichnis **data/database/projectiles** festgelegt. Im Tut habe ich die **Damage_Mine** verwendet. Öffnet man die gleichnamige txt-Datei so wird man auf den Command P_Explosion stoßen. Die 1 steht für die Anzahl.
- **this->P_CreateProjectile(decals_explosion, 1);** Erstellt ein Decal mit dem Command D_Decal der wieder im selbigen Verzeichnis mit Hilfe

einer Textdatei definiert ist. Die 1 steht für die Anzahl.

- **this->A_Play3DSound(dynamic, explosion_large, "");** Spielt den Sound einer großen Explosion.
- **this->PS_StartEffect(explosion_gasoline_huge, "");** Der PartikelEffekt für die Explosion

Ich hoffe nach dem Tut kann jeder seine eigenen Explosionen machen. Wer die Explosion noch echter wirken lassen will, kann noch ein kleines ruckeln einbauen. Dazu in der **DO_OnDeath**-Kategorie

player->cam_animateinplace(explosion); einfügen. Andere Kamerpfade finden sich in der txt-Datei im Verzeichnis **data/database/camerapaths**

© 2001 *rrr*bbb*, MPM