

[Sonntag, der 05.07.2009](#)

[Montag, der 06.07.2009](#)

[Dienstag, der 07.07.2009](#)

[Mittwoch, der 08.07.2009](#)

[Donnerstag, der 09.07.2009](#)

[Freitag, der 10.07.2009](#)

[Samstag, der 11.07.2009](#)

[Sonntag, der 12.07.2009](#)

[Montag, der 13.07.2009](#)

[Dienstag, der 14.07.2009](#)

[Mittwoch, der 15.07.2009](#)

[Donnerstag, der 16.07.2009](#)

[Freitag, der 17.07.2009](#)

[Samstag, der 18.07.2009](#)

[Donnerstag, der 30.07.2009](#)

[Sonntag, der 02.08.2009](#)

[Mittwoch, der 19.08.2009](#)

[Donnerstag, der 20.08.2009](#)

[Montag, der 21.09.2009](#)

[Sonntag, der 11.10.2009](#)

[Montag, der 12.10.2009](#)

[Montag, der 02.11.2009](#)

[Freitag, der 20.11.2009](#)

[Samstag, der 21.11.2009](#)

[Sonntag, der 25.04.2010](#)

[Montag, der 26.04.2010](#)

[Montag, der 05.07.2010](#)

[Dienstag, der 06.07.2010](#)

[Mittwoch, der 07.07.2010](#)

[Donnerstag, der 16.06.2011](#)

[Freitag, der 17.06.2011](#)

[Sonntag, der 26.06.2011](#)

[Donnerstag, der 30.06.2011](#)

Sonntag, der 05.07.2009

Nachdem ich in den letzten Tagen die Idee hatte, eine Art Left 4 Dead auf Basis von Gothic 2 zu erstellen, hab ich heute nun angefangen. Das Grundkonzept stand ja bereits durch Left 4 Dead: Man sollte sich einen von vier Charakteren wählen können und durch eine von Zombies bevölkerte Welt bis zu einem Zielpunkt gelangen. Also habe ich mir fix zwei Level in 3dsmax erstellt (Startbereich zur Konfiguration und das Testlevel, wo dann die Zombies kommen etc.) und meinen bereinigten Scriptordner zurechtgemacht.

In die zwei Level habe ich dann Triggerschleifen platziert, im Startlevel reicht eine sekundliche (wird nur zum Starten des Konfigurationsmenüs benutzt, siehe Code-Beispiel #1), im Testlevel auch die sekundliche Schleife und zusätzlich eine viel schnellere als präzisere Abfrage-Methode.

Code-Beispiel #1

```
if (CurrentLevel == STARTRAUM_ZEN)  
{  
  
if (Mod_InMenu == FALSE)  
{  
  
PLAYER_MOBSI_PRODUCTION = MOBSI_MONOLOG;  
  
Mod_InMenu = TRUE;  
  
AI_ProcessInfos (hero);  
  
};
```

```
};
```

Im Startlevel gibt es nun zusätzlich noch einen WP namens Testlevel, welcher von einem Levelchange umgeben ist, damit man beim Start über diesen WP ins Testlevel gelangt, und den Startpunkt für den Spieler. Das war es für das Startlevel auch schon wieder.

Im Testlevel gibt es einen Start-WP und eine Ziel-WP und zwischendrin noch einige, die derzeit nur zum Spawnen der Monster dienen (wird später über spezielle Spawn-Point für die einzelnen Monsterarten geregelt, ist nur im Testlevel jetzt anders).

Die Welten waren also erstmal fertig und ich konnte mich an die Konfiguration machen. Der Script dazu ist denkbar simpel, gleiches Prinzip wie schon in 63.

Code-Beispiel #2

```
INSTANCE Info_Mod_Hero_Start_Charakter (C_INFO)
```

```
{  
  
    npc          = PC_Hero;  
  
    nr           = 1;  
  
    condition    = Info_Mod_Hero_Start_Charakter_Condition;  
  
    information  = Info_Mod_Hero_Start_Charakter_Info;  
  
    permanent   = 0;  
  
    important    = 0;  
  
    description  = "Charakter wählen";  
  
};
```

```
FUNC INT Info_Mod_Hero_Start_Charakter_Condition ()
```

```
{  
  
    if (PLAYER_MOBSI_PRODUCTION == MOBSI_MONOLOG)  
        && (Mod_Charakter == 0)  
    {
```

return 1;

};

};

FUNC VOID Info_Mod_Hero_Start_Charakter_Info ()

{

Info_ClearChoices (Info_Mod_Hero_Start_Charakter);

*Info_AddChoice (Info_Mod_Hero_Start_Charakter, "Magie",
Info_Mod_Hero_Start_Charakter_Magie);*

*Info_AddChoice (Info_Mod_Hero_Start_Charakter, "Bogen",
Info_Mod_Hero_Start_Charakter_Bogen);*

*Info_AddChoice (Info_Mod_Hero_Start_Charakter, "Zweihand",
Info_Mod_Hero_Start_Charakter_Zweihand);*

*Info_AddChoice (Info_Mod_Hero_Start_Charakter, "Einhand",
Info_Mod_Hero_Start_Charakter_Einhand);*

};

FUNC VOID Info_Mod_Hero_Start_Charakter_Magie ()

{

Info_ClearChoices (Info_Mod_Hero_Start_Charakter);

Mod_Charakter = 4;

*B_SetNPCVisual (hero, MALE, "Hum_Head_Bald", Face_N_Milten, BodyTex_N,
ITAR_KDF_L);*

CreateInvItems (hero, ItMW_Addon_Stab01, 1);

CreateInvItems (hero, ItRu_IceCube, 1);

CreateInvItems (hero, ItRu_InstantFireball, 1);

};

FUNC VOID Info_Mod_Hero_Start_Charakter_Bogen ()

{

Info_ClearChoices (Info_Mod_Hero_Start_Charakter);

Mod_Charakter = 3;

B_SetNPCVisual (hero, FEMALE, "Hum_Head_Babe", FaceBabe_N_BlackHair, BodyTexBabe_N, ITAR_VlkBabe_H);

CreateInvItems (hero, ItRw_Bow_L_04, 1);

Mdl_SetModelFatness (hero, 0);

};

FUNC VOID Info_Mod_Hero_Start_Charakter_Zweihand ()

{

Info_ClearChoices (Info_Mod_Hero_Start_Charakter);

Mod_Charakter = 2;

B_SetNPCVisual (hero, MALE, "Hum_Head_Fighter", Face_B_Gorn, BodyTex_B, ITAR_SLD_H);

CreateInvItems (hero, ItMw_2h_Sld_Axe, 1);

};

FUNC VOID Info_Mod_Hero_Start_Charakter_Einhand ()

{

```
Info_ClearChoices (Info_Mod_Hero_Start_Charakter);
```

```
Mod_Charakter = 1;
```

```
B_SetNPCVisual (hero, MALE, "Hum_Head_Thief", Face_L_Diego, BodyTex_L,  
ITAR_DIEGO);
```

```
CreateInvItems (hero, ItMw_ShortSword5, 1);
```

```
};
```

```
INSTANCE Info_Mod_Hero_Start_Kampagne (C_INFO)
```

```
{
```

```
npc = PC_Hero;
```

```
nr = 2;
```

```
condition = Info_Mod_Hero_Start_Kampagne_Condition;
```

```
information = Info_Mod_Hero_Start_Kampagne_Info;
```

```
permanent = 0;
```

```
important = 0;
```

```
description = "Kampagne wählen";
```

```
};
```

```
FUNC INT Info_Mod_Hero_Start_Kampagne_Condition ()
```

```
{
```

```
if (PLAYER_MOBSI_PRODUCTION == MOBSI_MONOLOG)
```

```
&& (Mod_Kampagne == 0)
```

```
{
```

```
return 1;
```

```
};
```

```
};
```

FUNC VOID Info_Mod_Hero_Start_Kampagne_Info ()

{

Info_ClearChoices (Info_Mod_Hero_Start_Kampagne);

*Info_AddChoice (Info_Mod_Hero_Start_Kampagne, "Kampagne 1",
Info_Mod_Hero_Start_Kampagne_1);*

};

FUNC VOID Info_Mod_Hero_Start_Kampagne_1 ()

{

Info_ClearChoices (Info_Mod_Hero_Start_Kampagne);

Mod_Kampagne = 1;

ZombieHorden_Max = 2;

Tank_Max = 1;

};

INSTANCE Info_Mod_Hero_Start_EXIT (C_INFO)

{

npc = PC_Hero;

nr = 3;

condition = Info_Mod_Hero_Start_EXIT_Condition;

information = Info_Mod_Hero_Start_EXIT_Info;

permanent = 0;

important = 0;

description = "STARTEN";

};

```

FUNC INT Info_Mod_Hero_Start_EXIT_Condition ()
{
    if (PLAYER_MOBSI_PRODUCTION == MOBSI_MONOLOG)
        && (Mod_Charakter > 0)
        && (Mod_Kampagne > 0)
    {
        return 1;
    };
};

```

```

FUNC VOID Info_Mod_Hero_Start_EXIT_Info ()
{
    B_ENDPRODUCTIONDIALOG();

    if (Mod_Kampagne == 1)
    {
        AI_Teleport (hero, "TESTLEVEL");
    };
};

```

Wenn man die Kampagne wählt wird gleich festgelegt, wie viele Tanks und Zombiehorden maximal auf der Karte gespawnt werden dürfen (1 Tank und 2 Zombiehorden im Testlevel).

Gut, hat man also gestartet, findet man sich im Testlevel, zusammen mit den drei nicht gewählten Charakteren.

Code-Beispiel #3

```

FUNC VOID CharakterSpawnCenter ()

```



```

{
    if (Mod_Charakter != 4)
    {
        Wld_InsertNPC      (Char_Magie, "START");
    };
    if (Mod_Charakter != 3)
    {
        Wld_InsertNPC      (Char_Bogen, "START");
    };
    if (Mod_Charakter != 2)
    {
        Wld_InsertNPC      (Char_Zweihand, "START");
    };
    if (Mod_Charakter != 1)
    {
        Wld_InsertNPC      (Char_Einhand, "START");
    };
};

```

Die Funktion CharakterSpawnCenter() wird in der Startup jeder Kampagne aufgerufen und ist somit für jede Kampagne einsetzbar. Allgemein ist das ganze System flexibel angelegt, so dass man jederzeit später weitere Kampagnen hinzufügen kann.

Die Begleiter haben nun zwei TA's. Anfangs stehen sie am START-WP, sobald sich der Held bewegt folgen sie ihm (Dafür gibt's jetzt kein Beispiel, es sei denn, es wird ausdrücklich gewünscht).

Nun konnte ich also mit meinen drei Begleitern von einer Seite der Welt zur anderen Laufen (ziemlich unspektakulär ganz ohne Gegner). Also mussten Zombies her. Da habe ich zuerst testweise eine Zombiehorda in der Nähe des START-WP's eingefügt, um die Stärke der normalen Zombies zu skalieren. Danach gab's dafür eine eigene Funktion, die aber bei der richtigen Kampagne nochmal überarbeitet wird (oder auch vorher, mal sehen).

Code-Beispiel #4

FUNC VOID SpawnWave (var string WP)

```
{  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
    Wld_InsertNPC      (Zombie,    WP);  
  
};
```

FUNC VOID MonsterSpawner ()

```
{  
  
    // Monster spawnen  
  
  
    // Variable initialisieren  
  
  
    var int Randi;  
  
    Randi = Hlp_Random(100);  
  
};
```

// Wenn Variable im Bereich

if (Randi < 10)

{

// Wenn noch Zombiehorden übrig sind

if (ZombieHorden_Counter < ZombieHorden_Max)

{

// Wenn der Held noch weit genug vom Ziel entfernt ist

if (Npc_GetDistToWP(hero, "ZIEL") > 1000)

{

Randi = Hlp_Random(15);

// Wenn der Held am WP noch nicht vorbei ist und auch nicht zu nah dran

ist

if (Randi == 0)

&& (Mod_Was_A == FALSE)

&& (Npc_GetDistToWP(hero, "A") > 1000)

{

// Monster spawnen lassen

SpawnWave ("A");

// Counter hochsetzen

ZombieHorden_Counter += 1;

```
// WP abhaken

    Mod_Was_A = TRUE;
}
else if (Randi == 1)
    && (Mod_Was_B == FALSE)
    && (NPC_GetDistToWP(hero, "B") > 1000)
    {
        // Monster spawnen lassen

        SpawnWave ("B");

        // Counter hochsetzen

        ZombieHorden_Counter += 1;

        // WP abhaken

        Mod_Was_B = TRUE;
    }
else if (Randi == 2)
    && (Mod_Was_C == FALSE)
    && (NPC_GetDistToWP(hero, "C") > 1000)
    {
        // Monster spawnen lassen

        SpawnWave ("C");
```

```
// Counter hochsetzen

ZombieHorden_Counter += 1;

// WP abhaken

Mod_Was_C = TRUE;
}
else if (Randi == 3)
&& (Mod_Was_D == FALSE)
&& (Npc_GetDistToWP(hero, "D") > 1000)
{
// Monster spawnen lassen

SpawnWave ("D");

// Counter hochsetzen

ZombieHorden_Counter += 1;

// WP abhaken

Mod_Was_D = TRUE;
}
else if (Randi == 4)
&& (Mod_Was_E == FALSE)
&& (Npc_GetDistToWP(hero, "E") > 1000)
```

```
{  
  
    // Monster spawnen lassen  
  
    SpawnWave ("E");  
  
    // Counter hochsetzen  
  
    ZombieHorden_Counter += 1;  
  
    // WP abhaken  
  
    Mod_Was_E = TRUE;  
}  
  
else if (Randi == 5)  
&& (Mod_Was_F == FALSE)  
&& (Npc_GetDistToWP(hero, "F") > 1000)  
{  
  
    // Monster spawnen lassen  
  
    SpawnWave ("F");  
  
    // Counter hochsetzen  
  
    ZombieHorden_Counter += 1;  
  
    // WP abhaken  
  
    Mod_Was_F = TRUE;
```

```
}  
  
else if (Randi == 6)  
  
    && (Mod_Was_G == FALSE)  
  
    && (NPC_GetDistToWP(hero, "G") > 1000)  
  
    {  
  
        // Monster spawnen lassen  
  
  
        SpawnWave ("G");  
  
  
        // Counter hochsetzen  
  
  
        ZombieHorden_Counter += 1;  
  
  
        // WP abhaken  
  
  
        Mod_Was_G = TRUE;  
  
    }  
  
else if (Randi == 7)  
  
    && (Mod_Was_H == FALSE)  
  
    && (NPC_GetDistToWP(hero, "H") > 1000)  
  
    {  
  
        // Monster spawnen lassen  
  
  
        SpawnWave ("H");  
  
  
        // Counter hochsetzen  
  
  
        ZombieHorden_Counter += 1;
```

```
// WP abhaken

    Mod_Was_H = TRUE;
}
else if (Randi == 8)
    && (Mod_Was_I == FALSE)
    && (NPC_GetDistToWP(hero, "I") > 1000)
    {
        // Monster spawnen lassen

        SpawnWave ("I");

        // Counter hochsetzen

        ZombieHorden_Counter += 1;

        // WP abhaken

        Mod_Was_I = TRUE;
    }
else if (Randi == 9)
    && (Mod_Was_J == FALSE)
    && (NPC_GetDistToWP(hero, "J") > 1000)
    {
        // Monster spawnen lassen

        SpawnWave ("J");
```



```
// Counter hochsetzen

ZombieHorden_Counter += 1;

// WP abhaken

Mod_Was_J = TRUE;
}
else if (Randi == 10)
&& (Mod_Was_K == FALSE)
&& (Npc_GetDistToWP(hero, "K") > 1000)
{
// Monster spawnen lassen

SpawnWave ("K");

// Counter hochsetzen

ZombieHorden_Counter += 1;

// WP abhaken

Mod_Was_K = TRUE;
}
else if (Randi == 11)
&& (Mod_Was_L == FALSE)
&& (Npc_GetDistToWP(hero, "L") > 1000)
```

```
{  
  
    // Monster spawnen lassen  
  
    SpawnWave ("L");  
  
    // Counter hochsetzen  
  
    ZombieHorden_Counter += 1;  
  
    // WP abhaken  
  
    Mod_Was_L = TRUE;  
}  
  
else if (Randi == 12)  
&& (Mod_Was_M == FALSE)  
&& (Npc_GetDistToWP(hero, "M") > 1000)  
{  
  
    // Monster spawnen lassen  
  
    SpawnWave ("M");  
  
    // Counter hochsetzen  
  
    ZombieHorden_Counter += 1;  
  
    // WP abhaken  
  
    Mod_Was_M = TRUE;
```

```
}  
  
else if (Randi == 13)  
  
&& (Mod_Was_N == FALSE)  
&& (NPC_GetDistToWP(hero, "N") > 1000)  
  
{  
  
    // Monster spawnen lassen  
  
    SpawnWave ("N");  
  
    // Counter hochsetzen  
  
    ZombieHorden_Counter += 1;  
  
    // WP abhaken  
  
    Mod_Was_N = TRUE;  
}  
  
else if (Randi == 14)  
  
&& (Mod_Was_O == FALSE)  
&& (NPC_GetDistToWP(hero, "O") > 1000)  
  
{  
  
    // Monster spawnen lassen  
  
    SpawnWave ("O");  
  
    // Counter hochsetzen  
  
    ZombieHorden_Counter += 1;
```

```
// WP abhaken
```

```
Mod_Was_O = TRUE;
```

```
};
```

```
};
```

```
};
```

```
};
```

```
};
```

Diese Funktion wird in der ganz schnellen Schleife aufgerufen und gleich am Spielstart, damit eben die normalen Gruppen erstmal komplett platziert ist. Spezielle Trigger-Gruppen kommen auch noch (also Gruppen, die an speziellen Orten nochmal spawnen können durch irgendein Event).

Danach habe ich mich an den Tank gesetzt. Beim Tank handelt es sich um einen besonders kräftigen Gegner, der seine Gegner wegschleudert, viel Schaden macht und auch ordentlich einsteckt. Die Spawnfunktion ähnelt dabei der der Zombie-Horden.

Als nächstes habe ich mir die Kampfweise des Magiers vorgenommen. So hat man als Magier am Anfang zwei Zauber zur Verfügung, nämlich Eisblock und Feuerball. Gegen Bossmonster hat der Eisblock jedoch keine Wirkung und wird vom Magier-Begleiter auch nicht eingesetzt. Ebenso wird ein Gegner nur einmal eingefroren.

Code-Beispiel #5

```
func int B_SelectSpell (var C_NPC slf, var C_NPC oth)
```

```
{
```

```
    if (Hlp_GetInstanceID(slf) == Hlp_GetInstanceID(Char_Magie))
```

```
    {
```

```
        if (NPC_HasItems(slf, ItRu_FireRain) == 1)
```

```
        && (slf.attribute[ATR_MANA] >= SPL_Cost_FireRain)
```

```
        {
```

```
            B_ReadySpell (slf, SPL_FireRain, SPL_Cost_FireRain);
```

```

}

else if (NPC_HasItems(self, ItRu_IceWave) == 1)

&& (self.attribute[ATR_MANA] >= SPL_Cost_IceWave)

&& (!NPC_IsInState(oth, ZS_MagicFreeze))

&& (!C_IsBossMonster(oth))

{

    B_ReadySpell (self, SPL_IceWave, SPL_Cost_IceWave);

}

else if (NPC_HasItems(self, ItRu_IceCube) == 1)

&& (self.attribute[ATR_MANA] >= SPL_Cost_IceCube)

&& (!NPC_IsInState(oth, ZS_MagicFreeze))

&& (!C_IsBossMonster(oth))

{

    B_ReadySpell (self, SPL_IceCube, SPL_Cost_IceCube);

}

else if (NPC_HasItems(self, ItRu_InstantFireball) == 1)

&& (self.attribute[ATR_MANA] >= SPL_Cost_InstantFireball)

{

    B_ReadySpell (self, SPL_InstantFireball, SPL_Cost_InstantFireball);

}

else

{

    return FALSE;

};

return TRUE;

};

```

```
return FALSE; //alle anderen Gilden
```

```
};
```

Nun ist es so gewesen, dass die NPC's ziemlich schnell K.O. gegangen sind, deshalb mussten Tränke her. Jeder Charakter hat am Anfang einen Trank im Inventar. Hat er nun weniger als 25% seiner maximalen HP, benutzt er den Trank um sich zu heilen (wenn er Zeit dafür hat). Hat ein anderer Charakter weniger als 25% seiner maximalen HP und keinen Trank mehr, so bekommt er von einem anderen Charakter einen Trank und trinkt diesen dann auch. Natürlich kann auch der Held jedem anderen Charakter einen Trank geben, wenn dieser geschwächt ist, das funktioniert dann über den Hotkey "H". Wichtig war, dass immer nur ein Charakter einem anderen hilft.

Code-Beispiel #6

```
// Trankmanagement
```

```
// Held ist nicht Magier
```

```
if (Mod_Charakter != 4)
```

```
{
```

```
    // Selber versorgen wenn geschwächt und Trank dabei
```

```
        if (Char_Magie.attribute[ATR_HITPOINTS] <  
Char_Magie.attribute[ATR_HITPOINTS_MAX]/4)
```

```
            && (NPC_HasItems(Char_Magie, ItPo_Health) == 1)
```

```
            && (!NPC_IsInState(Char_Magie, ZS_Attack))
```

```
            && (!C_NpcIsDown(Char_Magie))
```

```
            && (!C_BodyStateContains(Char_Magie, BS_ITEMINTERACT))
```

```
{
```

```
        B_UseItem (Char_Magie, ItPo_Health);
```

```
};
```

```
    // Ansonsten die anderen checken (überhaupt dabei, geschwächt, wird noch nicht versorgt  
und hat auch keinen Trank)
```

```

if (Mod_Charakter != 3)

    && (Char_Bogen.attribute[ATR_HITPOINTS] <
Char_Bogen.attribute[ATR_HITPOINTS_MAX]/4)

    && (Char_Bogen.aivar[AIV_WirdVersorgt] == FALSE)

    && (NPC_HasItems(Char_Bogen, ItPo_Health) == 0)

    && (NPC_HasItems(Char_Magie, ItPo_Health) == 1)

    && (!NPC_IsInState(Char_Magie, ZS_Attack))

    && (!NPC_IsInState(Char_Bogen, ZS_Attack))

    && (!C_NpcIsDown(Char_Magie))

    && (!C_NpcIsDown(Char_Bogen))

    && (!C_BodyStateContains(Char_Magie, BS_ITEMINTERACT))

    && (!C_BodyStateContains(Char_Bogen, BS_ITEMINTERACT))

    {

        Char_Bogen.aivar[AIV_WirdVersorgt] = TRUE;

        AI_GotoNpc (Char_Magie, Char_Bogen);

        B_GiveInvItems (Char_Magie, Char_Bogen, ItPo_Health, 1);

        B_UseItem (Char_Bogen, ItPo_Health);

    };

```

```

if (Mod_Charakter != 2)

    && (Char_Zweihand.attribute[ATR_HITPOINTS] <
Char_Zweihand.attribute[ATR_HITPOINTS_MAX]/4)

    && (Char_Zweihand.aivar[AIV_WirdVersorgt] == FALSE)

    && (NPC_HasItems(Char_Zweihand, ItPo_Health) == 0)

    && (NPC_HasItems(Char_Magie, ItPo_Health) == 1)

    && (!NPC_IsInState(Char_Magie, ZS_Attack))

    && (!NPC_IsInState(Char_Zweihand, ZS_Attack))

```

```

&& (!C_NpcIsDown(Char_Magie))
&& (!C_NpcIsDown(Char_Zweihand))
&& (!C_BodyStateContains(Char_Magie, BS_ITEMINTERACT))
&& (!C_BodyStateContains(Char_Zweihand, BS_ITEMINTERACT))
{
    Char_Zweihand.aivar[AIV_WirdVersorgt] = TRUE;

    AI_GotoNpc (Char_Magie, Char_Zweihand);
    B_GiveInvItems (Char_Magie, Char_Zweihand, ItPo_Health, 1);
    B_UseItem (Char_Zweihand, ItPo_Health);
};

```

```

if (Mod_Charakter != 1)

```

```

    && (Char_Einhand.attribute[ATR_HITPOINTS] <
Char_Einhand.attribute[ATR_HITPOINTS_MAX]/4)
    && (Char_Einhand.aivar[AIV_WirdVersorgt] == FALSE)
    && (Npc_HasItems(Char_Einhand, ItPo_Health) == 0)
    && (Npc_HasItems(Char_Magie, ItPo_Health) == 1)
    && (!Npc_IsInState(Char_Magie, ZS_Attack))
    && (!Npc_IsInState(Char_Einhand, ZS_Attack))
    && (!C_NpcIsDown(Char_Magie))
    && (!C_NpcIsDown(Char_Einhand))
    && (!C_BodyStateContains(Char_Magie, BS_ITEMINTERACT))
    && (!C_BodyStateContains(Char_Einhand, BS_ITEMINTERACT))
{
    Char_Einhand.aivar[AIV_WirdVersorgt] = TRUE;

    AI_GotoNpc (Char_Magie, Char_Einhand);
    B_GiveInvItems (Char_Magie, Char_Einhand, ItPo_Health, 1);

```



```

        B_UseItem    (Char_Einhand, ItPo_Health);

};

if (hero.attribute[ATR_HITPOINTS] < hero.attribute[ATR_HITPOINTS_MAX]/4)

    && (hero.aivar[AIV_WirdVersorgt] == FALSE)

    && (NPC_HasItems(hero, ItPo_Health) == 0)

    && (NPC_HasItems(Char_Magie, ItPo_Health) == 1)

    && (!NPC_IsInState(Char_Magie, ZS_Attack))

    && (!C_NpcIsDown(Char_Magie))

    && (!C_NpcIsDown(hero))

    && (!C_BodyStateContains(Char_Magie, BS_ITEMINTERACT))

    && (!C_BodyStateContains(hero, BS_ITEMINTERACT))

    {

        hero.aivar[AIV_WirdVersorgt] = TRUE;

        AI_GotoNpc    (Char_Magie, hero);

        B_GiveInvItems    (Char_Magie, hero, ItPo_Health, 1);

        B_UseItem    (hero, ItPo_Health);

    };

};

```

Das war es dann auch schon für den ersten Tag am Projekt "Left 4 Gothic".

Montag, der 06.07.2009

Heute morgen ging es gleich weiter. Auf dem Programm stand heute das "Wiederbeleben". Wird man von einem Zombie besiegt, so stirbt man nicht, sondern wird bewusstlos. Das liegt man dann 30 Sekunden und muss wiederbelebt werden. Wenn man nicht wiederbelebt wird, dann stirbt man und ist endgültig tot.

Also los ging es. Das ganze war eigentlich recht einfach, sieht aber vielleicht ein bisschen kompliziert aus :D Das Grundprinzip funktioniert wie beim Trank-Management.

Code-Beispiel #1

```
// Wiederbelebungsmanagement

// Held ist nicht Magier

if (Mod_Charakter != 4)
{
    // Ansonsten die anderen checken (überhaupt dabei, geschwächt, wird noch nicht versorgt
und hat auch keinen Trank)

    if (Mod_Charakter != 3)
        && (Char_Bogen.aivar[AIV_WirdVersorgt] == FALSE)
        && (!NPC_IsInState(Char_Magie, ZS_Attack))
        && (NPC_IsInState(Char_Bogen, ZS_Unconscious))
        && (!C_NPCIsDown(Char_Magie))
        && (!C_BodyStateContains(Char_Magie, BS_ITEMINTERACT))
        && (!C_BodyStateContains(Char_Bogen, BS_ITEMINTERACT))
    {
        Char_Bogen.aivar[AIV_WirdVersorgt] = 4;

        AI_GotoNPC (Char_Magie, Char_Bogen);
        AI_PlayAni (Char_Magie, "T_PLUNDER");
        Char_Bogen.aivar[AIV_Versorgungszeit] = 0;
    }
};
```

```
if (Mod_Charakter != 2)

&& (Char_Zweihand.aivar[AIV_WirdVersorgt] == FALSE)

&& (!Npc_IsInState(Char_Magie, ZS_Attack))

&& (Npc_IsInState(Char_Zweihand, ZS_Unconscious))

&& (!C_NpcIsDown(Char_Magie))

&& (!C_BodyStateContains(Char_Magie, BS_ITEMINTERACT))

&& (!C_BodyStateContains(Char_Zweihand, BS_ITEMINTERACT))

{

    Char_Zweihand.aivar[AIV_WirdVersorgt] = 4;

    AI_GotoNpc (Char_Magie, Char_Zweihand);

    AI_PlayAni (Char_Magie, "T_PLUNDER");

    Char_Zweihand.aivar[AIV_Versorgungszeit] = 0;

};
```

```
if (Mod_Charakter != 1)

&& (Char_Einhand.aivar[AIV_WirdVersorgt] == FALSE)

&& (!Npc_IsInState(Char_Magie, ZS_Attack))

&& (Npc_IsInState(Char_Einhand, ZS_Unconscious))

&& (!C_NpcIsDown(Char_Magie))

&& (!C_BodyStateContains(Char_Magie, BS_ITEMINTERACT))

&& (!C_BodyStateContains(Char_Einhand, BS_ITEMINTERACT))

{

    Char_Einhand.aivar[AIV_WirdVersorgt] = 4;

    AI_GotoNpc (Char_Magie, Char_Einhand);

    AI_PlayAni (Char_Magie, "T_PLUNDER");

    Char_Einhand.aivar[AIV_Versorgungszeit] = 0;
```

```

};

if (hero.aivar[AIV_WirdVersorgt] == FALSE)
    && (!NPC_IsInState(Char_Magie, ZS_Attack))
    && (NPC_IsInState(hero, ZS_Unconscious))
    && (!C_NpcIsDown(Char_Magie))
    && (!C_BodyStateContains(Char_Magie, BS_ITEMINTERACT))
    && (!C_BodyStateContains(hero, BS_ITEMINTERACT))
{
    hero.aivar[AIV_WirdVersorgt] = 4;

    AI_GotoNpc (Char_Magie, hero);
    AI_PlayAni (Char_Magie, "T_PLUNDER");
    hero.aivar[AIV_Versorgungszeit] = 0;
};
};

```

Ursprünglich war es so, dass die NPC's direkt beim Aufruf dieser Funktion aufgestanden sind, bis dann die Versorgungszeit dazu kam. Damit muss der Helfer jetzt fünf Sekunden helfen, bis der Verletzte wirklich geheilt ist. Jetzt kommt es aber ja doch manchmal vor, dass man unterbrochen wird, sei es, dass man nochmal kämpfen muss, oder aber, dass man selber bewusstlos wird. Dafür habe ich dann den Versorgungsstatus zurücksetzen lassen. In der Funktion ist auch gleich die Verechnung der Dauer mit drin.

Code-Beispiel #2

```

if (Mod_Charakter != 4)
{
    if (NPC_IsInState(Char_Magie, ZS_Unconscious))
        && (Char_Magie.aivar[AIV_WirdVersorgt] > 0)
    {
        if (Char_Magie.aivar[AIV_WirdVersorgt] == 3)

```

```

{
    if (NPC_IsInState(Char_Bogen, ZS_Attack))
    || (C_NpcIsDown(Char_Bogen))
    {
        Char_Magie.aivar[AIV_WirdVersorgt] = 0;
        Char_Magie.aivar[AIV_Versorgungszeit] = 0;
    }
    else
    {
        if (NPC_GetDistToNPC(Char_Bogen, Char_Magie) < 200)
        && (CurrentTime > Char_Magie.aivar[AIV_LastVersorgt])
        {
            Char_Magie.aivar[AIV_Versorgungszeit] += 1;

            Char_Magie.aivar[AIV_LastVersorgt] = CurrentTime;

            if (Char_Magie.aivar[AIV_Versorgungszeit] == 5)
            {
                Char_Magie.aivar[AIV_Wiederbelebt] = TRUE;
            };
        };
    };
}

else if (Char_Magie.aivar[AIV_WirdVersorgt] == 2)
{
    if (NPC_IsInState(Char_Zweihand, ZS_Attack))
    || (C_NpcIsDown(Char_Zweihand))
    {

```

```

        Char_Magie.aivar[AIV_WirdVersorgt] = 0;

        Char_Magie.aivar[AIV_Versorgungszeit] = 0;

    }

    else

    {

        if (NPC_GetDistToNPC(Char_Zweihand, Char_Magie) < 200)
        && (CurrentTime > Char_Magie.aivar[AIV_LastVersorgt])
        {

            Char_Magie.aivar[AIV_Versorgungszeit] += 1;

            Char_Magie.aivar[AIV_LastVersorgt] = CurrentTime;

            if (Char_Magie.aivar[AIV_Versorgungszeit] == 5)
            {

                Char_Magie.aivar[AIV_Wiederbelebt] = TRUE;

            };

        };

    };

}

else if (Char_Magie.aivar[AIV_WirdVersorgt] == 1)

{

    if (NPC_IsInState(Char_Einhand, ZS_Attack))
    || (C_NPC_IsDown(Char_Einhand))
    {

        Char_Magie.aivar[AIV_WirdVersorgt] = 0;

        Char_Magie.aivar[AIV_Versorgungszeit] = 0;

    }

}

else

```

```

{
    if (NPC_GetDistToNPC(Char_Einhand, Char_Magie) < 200)
    && (CurrentTime > Char_Magie.aivar[AIV_LastVersorgt])
    {
        Char_Magie.aivar[AIV_Versorgungszeit] += 1;

        Char_Magie.aivar[AIV_LastVersorgt] = CurrentTime;

        if (Char_Magie.aivar[AIV_Versorgungszeit] == 5)
        {
            Char_Magie.aivar[AIV_Wiederbelebt] = TRUE;
        }
    };
};
}

else if (Char_Magie.aivar[AIV_WirdVersorgt] == 5)
{
    if (C_NPCIsDown(hero))
    {
        Char_Magie.aivar[AIV_WirdVersorgt] = 0;
        Char_Magie.aivar[AIV_Versorgungszeit] = 0;
    }
    else
    {
        if (NPC_GetDistToNPC(hero, Char_Magie) < 200)
        && (CurrentTime > Char_Magie.aivar[AIV_LastVersorgt])
        {
            Char_Magie.aivar[AIV_Versorgungszeit] += 1;

```


Dienstag, der 07.07.2009

Heute ging es erstmal gemütlich los. Zuerst im Forum die Beiträge beantwortet und währenddessen das Mod-Icon von Mantragon eingebaut.

Dann habe ich mich mal an das Item-Sammel-System gesetzt. So sollen die Begleiter ja selbstständig gefundene Items einsammeln, seien es nun neue Waffen oder Tränke (mehr ist ja noch nicht geplant). Damit aber nicht jeder Charakter jedes Item einsammelt und z.B. dann Diego mit Runen rumläuft, mit denen er gar nichts anfangen kann, musste das ein wenig spezialisiert werden. Dazu reicht aber auch eine einfache Abfrage. Ich hab die Funktion ein wenig umständlicher als nötig gemacht, dadurch ist es mir aber möglich, jede Waffe individuell abzufragen und somit Cheatern vorzubeugen (zumindest bringt es jetzt nichts, wenn man einfach die beste Zweihandwaffe reincheatet, auch wenn die gar nicht in der Mod benutzt wird, nur damit Gorn eben eine viel bessere Waffe hat, weil er die ja dann automatisch einsammelt).

Code-Beispiel #1

```
Npc_PerceiveAll (self);

if (Wld_DetectItem (self, ITEM_KAT_NF))
{
    if (Hlp_IsValidItem (item))
    && (Npc_GetDistToItem (self, item) < 1000)
    && (Npc_GetHeightToItem (self, item) < 230)
    {
        if (Hlp_GetInstanceID(self) == Hlp_GetInstanceID(Char_Einhand))
        && (Hlp_IsItem(item, ItMw_ShortSword5) == TRUE)
        && (Npc_HasItems(self, item) == 0)
        {
            self.aivar[AIV_ItemSammler] = FALSE;

            if (Npc_GetDistToItem (self, item) > 200)
            {
```

```

        AI_GotoItem (self, item);

};

    if (Hlp_IsValidItem(item))
    {
        AI_TakeItem (self, item);
    }

    Npc_SetStateTime (self, 0);
}

    else if (Hlp_GetInstanceID(self) == Hlp_GetInstanceID(Char_Zweihand))
    && (Hlp_IsItem(item, ItMw_2h_Sld_Axe) == TRUE)
    && (Npc_HasItems(self, item) == 0)
    {
        self.aivar[AIV_ItemSammler] = FALSE;

        if (Npc_GetDistToItem (self, item) > 200)
        {
            AI_GotoItem (self, item);
        }

        if (Hlp_IsValidItem(item))
        {
            AI_TakeItem (self, item);
        }

        Npc_SetStateTime (self, 0);
    }
}

    else

```

```

    {
        self.aivar[AIV_ItemSammler] = TRUE;

        NPC_SetStateTime (self, 0);
    };
}

```

Das ganze wird in der ZS_Follow_Player, also dem TA, in dem die Begleiter sind, durchlaufen.

Bei weiteren Testläufen ist mir dann aufgefallen, dass der Hero hin und wieder nicht wiederbelebt wurde. Das war ein Fehler, denn war Milten dafür eingetragen, den Held wiederzubeleben, ging es bei diesem nicht weiter mit den weiteren Abfragen => Held musste sterben.

Desweiteren gibt es nun eine Meldung, wenn jemand einen Trank von einem Gefährten erhält. Das sieht im Script so aus:

Code-Beispiel #2

```

func int B_GiveInvItems (var C_NPC giver, var C_NPC taker, var int itemInstance, var int amount)
{
    //FUNC

    // ----- Itmes transferieren -----

    NPC_RemoveInvItems (giver, itemInstance, amount);

    CreateInvItems          (taker, itemInstance, amount);

    // ----- PrintScreen -----

    var string concatText;

    var string itemname;

    itemname = item.name; //wird von CreateInvItems gesetzt --> Nico

```

```

if (Hlp_IsItem(item, ItPo_Health) == TRUE)
{
    if (Npc_IsPlayer(giver))
    {
        concatText = ConcatStrings("Du hast ", taker.name);
        concatText = ConcatStrings(concatText, " einen Heiltrank gegeben");
    }
    else if (Npc_IsPlayer(taker))
    {
        concatText = ConcatStrings("Du hast von ", giver.name);
        concatText = ConcatStrings(concatText, " einen Heiltrank bekommen");
    }
    else
    {
        concatText = ConcatStrings(giver.name, " hat ");
        concatText = ConcatStrings(concatText, taker.name);
        concatText = ConcatStrings(concatText, " einen Heiltrank gegeben");
    }
};

AI_PrintScreen    (concatText, -1, YPOS_GoldGiven, FONT_ScreenSmall, 2);

return TRUE;

};

```

Man bekommt also nicht nur eine Mitteilung, wenn man einen Trank erhält oder einen gibt, sondern auch, wenn die Begleiter untereinander Tränke austauschen, so kann man deren Trankpolitik ein wenig verfolgen.

Und die NPC's geben sich derzeit die Tränke ja noch über eine ziemlich große Entfernung. Das habe ich jetzt insofern geändert, dass ein Maximalabstand Bedingung für die Übergabe der Tränke ist. Somit ist es nicht mehr möglich, dass ein NPC 10 Meter von einem anderen entfernt steht und den Trank übergibt.

Nachdem die Story jetzt auf den vier Freunden basieren wird und der Spieler in die Rolle von einem dieser vier schlüpfen wird, wurde nun auch dessen Stimme angepasst, je nachdem, für wen er sich entscheidet. So wird gewährleistet, dass der Spieler dann auch mit der Stimme des gewählten Charakters die SVM's, also die Kommentare, die immer wieder auftauchen werden, spricht.

In diesem Zuge habe ich auch gleich einen Kommentar bei der Übergabe von Tränken eingebaut. So sagt jetzt jeder Charakter einen anderen, vom Inhalt aber gleichen Satz, wenn er jemandem einen Trank gibt. Natürlich noch ohne Sprachausgabe ;)

Jetzt gibt es auch noch Kommentare, wenn ein NPC einen Tank oder einen Boomer sieht (zumindest theoretisch, hab bisher die Dialogbox nie gesehen, aber die hat glaub ich auch nur eine geringe Reichweite, mit Sprachausgabe sollte es dann klappen).

Dann sollten nun Diego und Milten primär auf Boomer losgehen, wenn die in Sicht sind, damit die nicht zu nah an Lester und Gorn gelangen, um denen Schaden zuzufügen.

Und die Zauber Eiswelle und Feuerregen werden nun nicht mehr gecastet, wenn der Gegner viel zu weit weg ist. D.h. die Gegner müssen jetzt nah genug an Milten sein, damit er den Zauber einsetzt, bringt ja nichts, wenn er zaubert und letztlich niemanden trifft :D

Mantragon hat einen neuen Zombiekopf erstellt, aber leider ohne passenden Körper (und unterschiedliche Hautfarbe sieht schließlich doof aus :p). Momentan ist der mit Rüstung nur per Cheat ("insert Hordenfuehrer" in der Konsole im Marvin-Mode eintippen) zu betrachten.

Jetzt ist auch noch ThielHater zum Projekt gestoßen (wer ihn nicht kennt, er ist der Gründer und Leiter der [Gothic Reloaded Mod](#)). Er wird jetzt auch bei den Texturen helfen und kümmert sich derzeit um die Menütexturen und dergleichen. Eine erste Alpha-Version des Startscreens hat er mir schon gegeben für die heutige Version 0.1.2.

Zum Abschluss des heutigen Tages gab's ein neues, größeres und vor allem texturiertes Demolevel von

JohnPY. Die habe ich dann jetzt die letzte Stunde oder länger tauglich für die Mod gemacht und das neue Spawnsystem integriert. Jetzt gibt es auf der Map einen Tank, drei Boomer und Zombiehorden. Die werden alle zufällig auf der Map verteilt, allerdings an speziellen Spawnpoints für die jeweilige Zombieart. Außerdem wird an jedem Ort nur einmal gespawnt. Und das ganze wird jetzt durch die Startup aufgerufen und nicht mehr über die Schleife.

Und ich habe die Menütexe noch ein wenig angepasst, final sind die aber noch nicht. Desweiteren wurden die zwei Videos (Credits und Intro) aus dem Menü entfernt, haben schließlich nichts mit der Mod zu tun und sind daher überflüssig.

Das war es dann für die Version 0.1.2, die ich dann mal eben packe und dann hochlade und dann könnt ihr mal in einem neuen Level testen ;)

Mittwoch, der 08.07.2009

Heute ging es erst spät los, hab heute erst was für XR gemacht und dann jetzt kommt ein bisschen L4G.

Los ging es also mit der Öffentlichkeitsarbeit, ist bei dem Projekt ja das A und O, ist schließlich auf Präsentation usw. ausgelegt.

Unter den ganzen Foren-Beiträgen findet sich auch ein neuer Startraum von Mantragon. Der ist auch gleich bevobbt und beinhaltet einen Startpoint und einen Waypoint, also schon fast optimal. Musste lediglich noch den Waypoint umbenennen, einen Levelchange außen rum machen und die Triggerschleife einfügen und schon startet man in einem neuen Startraum (der diesmal auch wirklich ein Raum ist und nicht wieder eine Plane :D).

Nach ein paar Stunden anderer Beschäftigung (hauptsächlich privater Kram), hab ich nun mal ein Spawnsystem für Tränke eingebaut. Da gibt's im Demolevel 7 Spawnpoints und 3 Tränke werden gespawnt. Man kann noch wie vor nur einen Trank gleichzeitig dabei haben, sammelt man trotzdem einen ein, wird der gelöscht und man hat keine Rücklagen, falls man mal zurückkehrt um den Trank doch noch zu holen. Das Spawnsystem für Tränke sieht genauso aus wie das für die Zombies, es wird als zufällig einer der Spawnpoints aus der Liste ausgewählt und dort dann eben der Trank eingefügt. Dann wird der Spawnpoint abgehakt, damit da keine zwei Tränke übereinander stehen.

Code-Beispiel #1

```
Randi = Hlp_Random(PotionHealth_Spawnpoints_Levelpart_One_Max)+1;
```

```
if (Randi == 1)
&& (PotionHealth_Spawnpoints_Levelpart_One_01 == FALSE)
{
    Wld_InsertItem      (ItPo_Health, "FP_ITEM_POTION_HEALTH_01_01");

    PotionHealth_Spawnpoints_Levelpart_One_01 = TRUE;

    PotionHealth_Limit_Levelpart_One_Current += 1;
}
else if (Randi == 2)
&& (PotionHealth_Spawnpoints_Levelpart_One_02 == FALSE)
{
    Wld_InsertItem      (ItPo_Health, "FP_ITEM_POTION_HEALTH_01_02");

    PotionHealth_Spawnpoints_Levelpart_One_02 = TRUE;

    PotionHealth_Limit_Levelpart_One_Current += 1;
}
else if (Randi == 3)
&& (PotionHealth_Spawnpoints_Levelpart_One_03 == FALSE)
{
    Wld_InsertItem      (ItPo_Health, "FP_ITEM_POTION_HEALTH_01_03");

    PotionHealth_Spawnpoints_Levelpart_One_03 = TRUE;

    PotionHealth_Limit_Levelpart_One_Current += 1;
}
```

Ich habe jeden Kampagnenteil einzeln gegliedert, damit es nicht so ist, dass z.B. in Teil 1 der Kampagne alle Tank's der Kampagne sind und in Teil 3 z.B. erst Tränke auftauchen. Somit ist gewährleistet, dass überall so viele sind, wie man im Kampagnenauswahldialog einstellt.

Code-Beispiel #2

```
INSTANCE Info_Mod_Hero_Start_Kampagne (C_INFO)
```

```
{  
  
    npc          = PC_Hero;  
  
    nr           = 2;  
  
    condition    = Info_Mod_Hero_Start_Kampagne_Condition;  
  
    information  = Info_Mod_Hero_Start_Kampagne_Info;  
  
    permanent   = 0;  
  
    important    = 0;  
  
    description  = "Kampagne wählen";  
  
};
```

```
FUNC INT Info_Mod_Hero_Start_Kampagne_Condition ()
```

```
{  
  
    if (PLAYER_MOBSI_PRODUCTION == MOBSI_MONOLOG)  
        && (Mod_Kampagne == 0)  
    {  
        return 1;  
    };  
  
};
```

```
FUNC VOID Info_Mod_Hero_Start_Kampagne_Info ()
```

```
{  
  
    Info_ClearChoices (Info_Mod_Hero_Start_Kampagne);  
  
};
```



```
    Info_AddChoice    (Info_Mod_Hero_Start_Kampagne, "Kampagne 1",  
Info_Mod_Hero_Start_Kampagne_1);
```

```
};
```

```
FUNC VOID Info_Mod_Hero_Start_Kampagne_1 ()
```

```
{
```

```
    Info_ClearChoices    (Info_Mod_Hero_Start_Kampagne);
```

```
    Mod_Kampagne = 1;
```

```
    ZombieHorden_Max = 2;
```

```
    Tank_Max = 1;
```

```
    Boomer_Max = 1;
```

```
    // Level #1 Limits setzen
```

```
    Tank_Spawnpoints_Levelpart_One_Max = 6;
```

```
    Tank_Limit_Levelpart_One_Max = 1;
```

```
    Boomer_Spawnpoints_Levelpart_One_Max = 10;
```

```
    Boomer_Limit_Levelpart_One_Max = 3;
```

```
    Zombiehorde_Spawnpoints_Levelpart_One_Max = 24;
```

```
    Zombiehorde_Limit_Levelpart_One_Max = 5;
```

```
    PotionHealth_Spawnpoints_Levelpart_One_Max = 7;
```

```
    PotionHealth_Limit_Levelpart_One_Max = 3;
```

```
    // Level #2 Limits setzen
```

```
    Tank_Spawnpoints_Levelpart_Two_Max = 0;
```

```
    Tank_Limit_Levelpart_Two_Max = 0;
```

Boomer_Spawnpoints_Levelpart_Two_Max = 0;

Boomer_Limit_Levelpart_Two_Max = 0;

Zombiehorde_Spawnpoints_Levelpart_Two_Max = 0;

Zombiehorde_Limit_Levelpart_Two_Max = 0;

PotionHealth_Spawnpoints_Levelpart_Two_Max = 0;

PotionHealth_Limit_Levelpart_Two_Max = 0;

// Level #3 Limits setzen

Tank_Spawnpoints_Levelpart_Three_Max = 0;

Tank_Limit_Levelpart_Three_Max = 0;

Boomer_Spawnpoints_Levelpart_Three_Max = 0;

Boomer_Limit_Levelpart_Three_Max = 0;

Zombiehorde_Spawnpoints_Levelpart_Three_Max = 0;

Zombiehorde_Limit_Levelpart_Three_Max = 0;

PotionHealth_Spawnpoints_Levelpart_Three_Max = 0;

PotionHealth_Limit_Levelpart_Three_Max = 0;

// Level #4 Limits setzen

Tank_Spawnpoints_Levelpart_Four_Max = 0;

Tank_Limit_Levelpart_Four_Max = 0;

Boomer_Spawnpoints_Levelpart_Four_Max = 0;

Boomer_Limit_Levelpart_Four_Max = 0;

Zombiehorde_Spawnpoints_Levelpart_Four_Max = 0;

Zombiehorde_Limit_Levelpart_Four_Max = 0;

PotionHealth_Spawnpoints_Levelpart_Four_Max = 0;

PotionHealth_Limit_Levelpart_Four_Max = 0;

};

Die alten Variablen `ZombieHorden_Max`, `Tank_Max` und `Boomer_Max` sind noch Relikte vom alten Spawnsystem und sind mittlerweile überflüssig. Da das Demolevel als Levelteil 1 zählt, sind die übrigen Level (momentan bis 4) alle auf 0 gesetzt. Man könnte das so schon nutzen für eine komplette Welt.

"`Tank_Spawnpoints_Levelpart_One_Max`" ist die Variable, in der festgelegt wird, wie viele Spawnpoints für Tanks es im Levelteil 1 gibt. Aus diesen Spawnpoints wird dann eben einer ausgesucht.

"`Tank_Limit_Levelpart_One_Max`" ist die Variable, mit der festgelegt wird, wie viele Tanks in dem Levelteil auftauchen sollen. Wo die sind, ist dann eben Zufall. Könnten auf der Map eben max. 6 sein (setzt man die Limit-Variable auf einen Wert größer 6 crasht die Kampagne beim Start).

Aufgrund der teilweise geringen Trefferwahrscheinlichkeit von Magie, wenn die Gegner zu nah am Magier dran sind, wird Milten nun in diesem Fall auf Magie (Feuerball und Eisblock) verzichten und stattdessen seinen Stab benutzen. Wäre ja sonst Manaverschwendung und das ist in Zeiten von Zombiehorden ziemlich rar ;)

Wegen der Performance-Probleme bei einige Usern habe ich mal die Triggerschleife deutlich langsamer durchlaufen lassen. Ich hoffe, dass das was bringt. Ob, erfahren wir durch das Feedback nach der neuen Version, die jetzt kommt :)

Donnerstag, der 09.07.2009

Auf ans Werk, los geht es mit den Kritikpunkten im Forum.

Da wäre zuerst der fehlende Flächenschaden des Boomers, bei seinem Ableben. Den gibt's zwar schon, aber die Reichweite ist vielleicht zu gering (wobei es ja trotzdem 3 Meter sind). Hab das mal auf 5 Meter hochgesetzt, mal sehen, ob das dann passt.

Dann wurde noch kritisiert, dass der Boomer so schnell stirbt. Also habe ich den ein wenig robuster gemacht.

Dann wurde vorgeschlagen, dass Diego noch eine Nahkampfwaffe bekommen sollte. In diesem Zuge habe ich die kompletten Waffen und Runen für alle Charaktere erstellt und vom Schaden her angeglichen. Jedem stehen nun für die Primärklasse 6 verschiedene Waffen zur Verfügung (eine hat man am Anfang und für jeden Level der Kampagne eine bessere). Für die Sekundärwaffe bei Diego und Milten gibt's

neben der Startwaffe nur noch zwei bessere, die dann noch deutlich schwächer als die Primärwaffen sind. Bei Milten ist der doch ziemlich überflüssige Eisblock rausgeflogen.

Auch auf dieser Änderung beruhend sammeln die Begleiter nun alle Waffen ein, die für sie bestimmt sind. Und sie sollten neue Waffen jetzt auch gleich anlegen und nicht nur im Inventar behalten.

So, nun hat Diego keinen Pfeilrespawn mehr und auch der Spieler natürlich nicht mehr, wenn er Diego spielt. Zu Beginn hat man jetzt immer 200 Pfeile (erstmal), in den Köchern werden dann entweder 50 oder 100 drin sein, denke ich mal.

Als nächstes habe ich erstmal was anderes gemacht, schließlich hatte ich noch einige Texturen. Also habe ich die Zombiefrau von Mantragon mal eingebaut und die spawnt jetzt auch mit den anderen Zombies zusammen und ist ein fieser Gegner (so wie die anderen halt :D). Und ThielHater hat einen ganzen Schwung Texturen für das Menü fertig. Schaut sie euch am besten selbst an, die jetzt alle aufzulisten ist mir grad zu aufwendig ;) Weibliche Zombies sind übrigens keine wirkliche Neuerung, in meinem Hauptprojekt "Xeres' Rückkehr" gibt es die auch schon :)

Nächster Kritikpunkt war ja die zu geringe Sinnesreichweite der Zombies, wodurch es einem als Diego und Milten möglich war, die Zombies alle zu töten, bevor die reagiert haben. Das geht nun nicht mehr, denn jetzt ist die Reichweite so groß, dass man dazu keine Zeit mehr hat (ihr werdet sehen).

Dann habe ich mal die erste Version des Menümusik-Stückes von Jim Hal Wilson aus meinem Team (Teutonic Team, Gothic II Mod "Xeres' Rückker") eingebaut. Auch die ist in der 0.1.4 enthalten. Dazu ist übrigens eure Meinung und Kritik gefragt.

Um noch ein wenig was produktives zu machen, gehe ich die Wiederbelebungsproblematik an. So ist es derzeit so, dass zwar jeder NPC nur von einem geheilt wird, aber nicht einer nur einen gleichzeitig wiederbeleben kann. Da das nicht ganz optimal ist, werden auch die Wiederbeleber jetzt ein Aivar bekommen, wo geregelt wird, ob die schon wiederbeleben und wenn ja, dann eben nur einen. Das aivar heißt nun AIV_Belebt, also er belebt gerade jemanden wieder :) Übrigens gehen mir langsam die Aivar-Plätze aus, muss mich dann wohl mal ans auskommentieren der alten Aivars machen, damit ich da ein paar Plätze frei bekomme, aber erstmal sehen, wie viele ich noch so brauche.

Der nächste Punkt war das Entfernen der Effekte beim Sterben der Zombies. Dies hat natürlich bei vielen zu starken Performance-Problemen geführt. Deswegen wurden die PFX-Effekte nun entfernt.

Da immer die gleiche Anzahl an Zombies natürlich auf Dauer ein wenig abwechslungsarm ist, ist die Größe der Zombiehorden nun Variabel. Allerdings sind es immer min. 10 Zombies, die maximale Limitierung liegt erstmal bei 20. Die Funktion dazu wird in der SpawnWave aufgerufen.

Wld_InsertNpc (Zombie, WP);

Wld_InsertNpc (Zombie, WP);

var int Randi;

Randi = Hlp_Random(10)+1;

SpawnZusatzWave (WP, Randi);

};

Was mir bei meinen Tests noch so aufgefallen ist, ist, dass die Tanks manchmal durch den Flugschaden trotzdem töten, statt nur ohnmächtig werden zu lassen. Auch dieses Problem sollte nun behoben sein.

Das war es dann für die 0.1.4 und wahrscheinlich auch für heute, es sei denn irgendwer gibt mir heute noch Material zum Einbauen :p

Nachdem mir jetzt bei einem kurzen Test der Version 0.1.4 aufgefallen ist, dass ich meine Kameraden als Milten mit dem Kampfstab verletzen konnte, habe ich das natürlich gleich noch geändert, nicht, dass ich das noch vergesse und sich das ewig durch die einzelnen Version zieht :D

Freitag, der 10.07.2009

Neuer Tag, neues Glück.

Bevor ich die neuen Beiträge im Forum beantworte ändere ich noch schnell die Startzeit im Testlevel: Da spielt man nun also ab 20 Uhr und es ist gleich ein wenig dunkler :D

Dann mal auf an die Beiträge.

Diego hat zu viele Pfeile. Letztlich eine Sache des Balancings, aber ich habe es mal runtergesetzt auf 50 Pfeile.

Die nächste Neuerung ist wohl eher als Easteregg anzusehen, andererseits auch nicht unbedingt. Ich sag jetzt nicht, was es ist, aber bleibt einfach mal eine Weile regungslos stehen ;) (erstmal so fünf Minuten und dann nochmal zwei Minuten, nach den weiteren zwei Minuten passiert aber (noch) nicht immer was).

Was Rohrschach ja noch erwähnt hatte, war das Einfügen der Fähigkeiten der Charaktere im Auswahlbildschirm. Das gibt es jetzt auch.

Als nächstes kommt jetzt die Ausarbeitung der Story. Die wird nicht ins Forum gestellt, damit nicht die ganze Spannung genommen wird. Ob man die im vornherein spielen kann, weiß ich noch nicht genau, auf jedenfall wird man ein paar Screens zu sehen bekommen, wenn die Welten dazu stehen ;)

Nun habe ich noch den Köcher von Mantragon eingebaut, inklusive Spawn. Gibt aber nur zwei Spawnpoints und einen Köcher im Level, aber Diego hat ja ohnehin schon genug Pfeile zur Zeit :D

Damit noch was Neues in der 0.1.5 enthalten ist, habe ich jetzt noch einen Kommentar eingebaut, wenn man wiederbelebt wurde. Also der Wiederbelebte gibt jetzt immer einen Kommentar ab.

Samstag, der 11.07.2009

Damit auch diejenigen die Mod spielen können, die eine etwas schwächere CPU haben, gibt es jetzt im Konfigurationsmenü am Anfang die Option die CPU-Leistung auszuwählen. Hat man bei den bisherigen Versionen keine Probleme gehabt, so kann man die schnelle CPU wählen, ist es jedoch wie bei Rohrschach unspielbar gewesen, sollte man die langsame CPU wählen.

Dann gibt es für die, die sich nicht entscheiden können, mit welchem Charakter sie spielen wollen, eine Zufalls-Charakterwahl.

Code-Beispiel #1

```
var int Randi;
```

```
Randi = Hlp_Random(4)+1;
```

```
if (Randi == 1)
```

```

{
    Info_Mod_Hero_Start_Charakter_Einhand();
}
else if (Randi == 2)
{
    Info_Mod_Hero_Start_Charakter_Zweihand();
}
else if (Randi == 3)
{
    Info_Mod_Hero_Start_Charakter_Bogen();
}
else if (Randi == 4)
{
    Info_Mod_Hero_Start_Charakter_Magie();
};

```

ThielHater hat heute zwei neue Texturen gemacht. Die eine ist für den Tank und die andere für die Witch. Da ist ein Prototyp enthalten, die Kampfaktik, Balancing und Spawning fehlt noch, kommt dann wahrscheinlich morgen. Wer die aktuelle Witch mal testen will, in der Konsole "insert witch" eingeben ;)

Von Jim Hal Wilson gab es auch noch mal eine verbesserte Version des Menü-Stückes.

Der heutige Changelog ist sogar noch kürzer, als der von gestern, aber ich hatte heute einfach nicht so viel Zeit. Sollte die nächsten Tage wieder ein wenig besser werden und dann wird wohl auch mal das Mapping anlaufen, wenn sich genug freiwillige finden :)

Sonntag, der 12.07.2009

Zuerst habe ich heute dafür gesorgt, dass ein wiederbelebender Gefährte nur noch durch Angriffe davon abgehalten wird. Das sollte dazu führen, dass nicht die Nähe eines Gegner schon dafür sorgt, dass die

Wiederbelebung abgebrochen wird. Schließlich braucht man für die Kämpfe möglichst jeden und solange der Gegner noch nicht nah genug ist, kann man also noch wiederbeleben.

Dann sollten die vier Charaktere nun nach 20 Sekunden rumliegen einen Kommentar loslassen, schließlich wird die Zeit da dann langsam eng.

Dann wird jetzt noch eine Witch gespawnt. Die ignoriert die Überlebenden nicht, sondern ist sehr offensiv. Momentan kommt sie aber noch nicht so oft zum Zuge, da muss ich sehen, wie sich das beheben lässt.

Von ThielHater gab es nochmal neue Texturen für Tank und Witch, welche natürlich auch beide eingebaut sind.

Ich habe dann jetzt noch eine Fogzone und Plane gesetzt und den Himmel aus dem Minental übernommen (wobei man den voraussichtlich nicht sehen wird, wenn man nie tagsüber spielt :D).

Außerdem habe ich die fehlende Zombie.msb ins Modvolumen gepackt, hatte das wohl vergessen, deswegen habt ihr noch Effekte beim Sterben der Zombies gehabt.

Montag, der 13.07.2009

Neuerung #1: Die Witch kündigt sich jetzt bei jedem Teleport durch einen Schrei an :D Bei den Schreien handelt es sich um die der Harpien.

Dann gibt's jetzt noch für jeden Charakter einen weiteren Spruch, wenn er am Boden liegt und auf Wiederbelebung wartet. Da kommen sicher noch mehr ;)

Außerdem halten sich jetzt alle Nahkämpfer zurück, bis die Gegner in Nahkampfreichweite sind. Das gilt anfangs nur für Lester und Gorn, später, wenn Pfeile bzw. Mana alle sind auch für Diego und Milten. Dann kämpfen alle vier zusammen im Nahkampf. Funktioniert gegen normale Zombies ausgesprochen gut die Taktik :D

Dann hat sich Lees Rechte Hand eine "Punktliste" gewünscht. Die gibt es jetzt und somit kann man sehen, wer der fleißigste Metzler ist §ugly Normale Zombies bringen einen Punkt, Bosszombies 5.

Mir ist aufgefallen, dass Milten im Vergleich zu Diego viel weniger Schaden macht. Das liegt daran, dass Diego ja immer Volltreffer macht (also Waffenschaden + Stärke usw.), während Milten nur den Schaden des Zaubers verursacht. Deswegen habe ich den Schaden der Zauber erhöht, um das einigermaßen auszugleichen.

Das gemeldete Problem, das Gorn einem manchmal nicht folgt, sollte jetzt auch behoben sein. Dazu wurden die von Anfang an in den Verfolgungsmodus versetzt.

Dienstag, der 14.07.2009

Munter geht's weiter und los geht's heute mit einem Kommentar (bzw. eigentlich sieben), warum das Trank geben nicht klappt. Da erklärt der Spieler dem Spieler (klingt komisch, ist aber so), warum er dem Ziel keinen Trank geben kann, sei es nun, weil der noch einen Trank hat, weil er noch zu viele HP hat, weil man selber keinen Trank mehr oder halt die anderen Sachen :D Wenn bei euch die Meldung allerdings immer noch nicht erscheinen sollte (einfach gleich am Anfang mal einen anderen Charakter mit H anklicken), dann stimmt da was nicht.

Dann haben die Charaktere noch zwei Sprüche bekommen, wenn sie einen Boss getötet haben. Dabei ist es egal, was für ein Boss es war und die zwei Sprüche sind momentan noch bei allen gleich.

Desweiteren gibt es jetzt eine Sieg-Meldung, wenn man alle Zombies getötet hat. Da aber manchmal die Gegner erst später auftauchen (warum auch immer), kommt die Meldung eben nicht immer.

Ein neuer Kritikpunkt war jetzt, dass alle mit den gleichen Kampfanis kämpfen. So haben Diego und Milten jetzt die Stufe 2 - Nahkampf-Anis, während Gorn und Lester noch auf Stufe 3 bleiben. Außerdem ist jetzt das Balancing ein wenig verändert. Gorn ist robuster und stärker als die anderen, Lester immer noch robuster und stärker als die Fernkämpfer, hält aber nicht mehr so viel aus wie Gorn. Dann kommt Diego und dann Milten, der als Magier natürlich nicht so viel aushält, dafür aberm vor allem später dann, mit seinem Flächenschaden ordentlich punkten wird (hoffentlich).

Damit auch noch was ganz neues drin ist, gibt es jetzt den Hunter. Der ist zwar erstmal nur ein Prototyp, sollte aber schon gefährlich sein. Er hat neun Spawnpoints und taucht zweimal auf.

Was mir noch so aufgefallen ist bei diversen Tests ist, dass die Bossgegner immer zu ihr Ziel wechseln und somit nie so wirklich zum Schlag kommen. Das ist jetzt auch anders und die bleiben bei ihrem Gegner, bis sie tot oder der Gegner down ist. Natürlich wird jetzt nicht immer der Spieler das Ziel bleiben, nur weil er voran geht und somit als Ziel prädestiniert ist. Das wird jetzt zufällig ausgelost,

welchen Charakter die Bosse angreifen :)

Jetzt habe ich eine aufwendigere Änderung von vorhin rückgängig gemacht und die normalen Zombies greifen auch nach dem Zufallsprinzip die Überlebenden an. D.h. jetzt wird auf jedenfall nicht mehr nur einer von bis zu 20 Zombies auf einmal angegriffen, sondern ein Grüppchen auf Lester, eins auf Gorn usw.

Mittwoch, der 15.07.2009

Da es wohl vorkommt, dass Hunter und Tank sich bekämpfen, habe ich das jetzt natürlich geändert.

Dann habe ich den Hunter noch etwas schwächer gemacht, weil er doch ein wenig zu robust war.

Donnerstag, der 16.07.2009

Damit es etwas Neues gibt heute, habe ich mich mal an die Manatränke gesetzt (bzw. den Manatrank). Der regeneriert 25% des maximalen Mana von Milten und ist im Testlevel zweimal zu finden. Dafür wurde das maximale Mana von Milten runtergesetzt, sonst ist der ja voll stark, das geht ja mal gar nicht :D

Dann habe ich den Flächenschaden beim Ableben des Boomers wieder reaktiviert, nachdem das ja nicht der Grund für die Stilllegung des Counters war (hab ich gestern vergessen).

Momentan war es ja so, dass Heiltränke und Köcher, die man unberechtigt eingesammelt hat (also wenn man schon einen Heiltrank hatte oder bei den Köchern, wenn man nicht Diego war), einfach entfernt wurden und somit aus dem Spiel waren. Jetzt ist es so, dass die gedroppt werden und somit den anderen noch zur Verfügung stehen.

Code-Beispiel #1

// Tränke entfernen und Meldung Printen, wenn zu viele Tränke gesammelt

```
if (Npc_HasItems(hero, ItPo_Health) == 0)
```

```

&& (Mod_Spawn_Trunk == 1)
{
    CreateInvItems    (hero, ItPo_Health, 1);

    Mod_Spawn_Trunk = 0;
};

if (Npc_HasItems(hero, ItPo_Health) >= 2)
{
    AI_DropItem (hero, ItPo_Health);

    Mod_Spawn_Trunk = 1;

    Print ("Du kannst nur einen Trank haben!");
};

```

Die Variable dient zum erstatten des einen Trankes, denn beim droppen werden alle gedroppt (zwar nur als ein Item, also wenn man es wieder aufnimmt, hat man nur einen Trank, aber hatte man zwei, verschwinden beide aus dem Inventar). Durch die Variable wird also wieder ein Trank ins Inventar eingefügt, sobald die überflüssigen gedroppt wurden.

Freitag, der 17.07.2009

Auch wenn ich heute Geburtstag habe, habe ich natürlich Freizeit, die in die Mod fließt :)

Daher habe ich mal fix die Printmeldungen beim Einsammeln von einem zweiten Heiltrank oder einem Köcher als Nicht-Diego als SVM gemacht und die Charaktere sagen jetzt was dazu, statt das die Printmeldung kommt.

Dann gibt's jetzt endlich einen Schwierigkeitsregler. Der verändert noch nicht viel: Auf Leicht dauert es

15 Sekunden länger, bis man stirbt, wenn man bewusstlos ist, und auf Schwer spawnen bis zu 10 Zombies pro Zombiehorde mehr. Auf Normal ist alles geblieben.

Code-Beispiel #1

```
INSTANCE Info_Mod_Hero_Start_Schwierigkeitsgrad (C_INFO)
```

```
{  
  
    npc          = PC_Hero;  
  
    nr          = 3;  
  
    condition    = Info_Mod_Hero_Start_Schwierigkeitsgrad_Condition;  
  
    information   = Info_Mod_Hero_Start_Schwierigkeitsgrad_Info;  
  
    permanent    = 0;  
  
    important    = 0;  
  
    description  = "Schwierigkeitsgrad wählen";  
  
};
```

```
FUNC INT Info_Mod_Hero_Start_Schwierigkeitsgrad_Condition ()
```

```
{  
  
    if (PLAYER_MOBSI_PRODUCTION == MOBSI_MONOLOG)  
  
    {  
  
        return 1;  
  
    };  
  
};
```

```
FUNC VOID Info_Mod_Hero_Start_Schwierigkeitsgrad_Info ()
```

```
{  
  
    Info_ClearChoices (Info_Mod_Hero_Start_Schwierigkeitsgrad);  
  
    Info_AddChoice (Info_Mod_Hero_Start_Schwierigkeitsgrad, "Schwer",  
Info_Mod_Hero_Start_Schwierigkeitsgrad_Schwer);  
  
};
```

```
    Info_AddChoice (Info_Mod_Hero_Start_Schwierigkeitsgrad, "Normal",  
Info_Mod_Hero_Start_Schwierigkeitsgrad_Normal);
```

```
    Info_AddChoice (Info_Mod_Hero_Start_Schwierigkeitsgrad, "Leicht",  
Info_Mod_Hero_Start_Schwierigkeitsgrad_Leicht);
```

```
};
```

```
FUNC VOID Info_Mod_Hero_Start_Schwierigkeitsgrad_Schwer ()
```

```
{
```

```
    Info_ClearChoices (Info_Mod_Hero_Start_Schwierigkeitsgrad);
```

```
    Mod_Schwierigkeitsgrad = 3;
```

```
    HAI_TIME_UNCONSCIOUS = 30;
```

```
    ZombieHorden_Max = 20;
```

```
};
```

```
FUNC VOID Info_Mod_Hero_Start_Schwierigkeitsgrad_Normal ()
```

```
{
```

```
    Info_ClearChoices (Info_Mod_Hero_Start_Schwierigkeitsgrad);
```

```
    Mod_Schwierigkeitsgrad = 2;
```

```
    HAI_TIME_UNCONSCIOUS = 30;
```

```
    ZombieHorden_Max = 10;
```

```
};
```

```
FUNC VOID Info_Mod_Hero_Start_Schwierigkeitsgrad_Leicht ()
```

```

{
    Info_ClearChoices (Info_Mod_Hero_Start_Schwierigkeitsgrad);

    Mod_Schwierigkeitsgrad = 1;

    HAI_TIME_UNCONSCIOUS = 45;

    ZombieHorden_Max = 10;
};

```

Samstag, der 18.07.2009

Nachdem mir nichts Besseres eingefallen ist, habe ich mich halt heute an den Spawn der Waffen gesetzt. Es gibt im Testlevel für alle Primärwaffen 3 Spawnpoints und für die Sekundärwaffen von Diego und Milten 2. In der Kampagne werden die natürlich besser versteckt sein ;) Die Waffen werden natürlich vom entsprechenden Träger automatisch eingesammelt und ausgerüstet.

Code-Beispiel #1

```

FUNC VOID WeaponSpawner ()
{
    var int Randi;

    // Primärwaffen

    // Levelpart One

    Randi = Hlp_Random(Weapon_Lester_Level_01)+1;

    Wld_InsertItem (ItMw_Lester_02, ConcatStrings("FP_ITEM_LESTER_01_0",

```

```
IntToString(Randi)));
```

```
Randi = Hlp_Random(Weapon_Gorn_Level_01)+1;
```

```
Wld_InsertItem (ItMw_Gorn_02, ConcatStrings("FP_ITEM_GORN_01_0",  
IntToString(Randi)));
```

```
Randi = Hlp_Random(Weapon_Diego_Level_01)+1;
```

```
Wld_InsertItem (ItRw_Diego_02, ConcatStrings("FP_ITEM_DIEGO_01_0",  
IntToString(Randi)));
```

```
Randi = Hlp_Random(Weapon_Milten_Level_01)+1;
```

```
Wld_InsertItem (ItRu_Milten_02, ConcatStrings("FP_ITEM_MILTEN_01_0",  
IntToString(Randi)));
```

```
// Levelpart Two
```

```
if (Weapon_Lester_Level_02 > 0)
```

```
{
```

```
Randi = Hlp_Random(Weapon_Lester_Level_02)+1;
```

```
Wld_InsertItem (ItMw_Lester_03, ConcatStrings("FP_ITEM_LESTER_02_0",  
IntToString(Randi)));
```

```
};
```

```
if (Weapon_Gorn_Level_02 > 0)
```

```
{
```

```
Randi = Hlp_Random(Weapon_Gorn_Level_02)+1;
```



```
Wld_InsertItem (ItMw_Gorn_03, ConcatStrings("FP_ITEM_GORN_02_0",  
IntToString(Randi)));
```

```
};
```

```
if (Weapon_Diego_Level_02 > 0)
```

```
{
```

```
Randi = Hlp_Random(Weapon_Diego_Level_02)+1;
```

```
Wld_InsertItem (ItRw_Diego_03, ConcatStrings("FP_ITEM_DIEGO_02_0",  
IntToString(Randi)));
```

```
};
```

```
if (Weapon_Milten_Level_02 > 0)
```

```
{
```

```
Randi = Hlp_Random(Weapon_Milten_Level_02)+1;
```

```
Wld_InsertItem (ItRu_Milten_03, ConcatStrings("FP_ITEM_MILTEN_02_0",  
IntToString(Randi)));
```

```
};
```

```
// Levelpart Three
```

```
if (Weapon_Lester_Level_03 > 0)
```

```
{
```

```
Randi = Hlp_Random(Weapon_Lester_Level_03)+1;
```

```
Wld_InsertItem (ItMw_Lester_04, ConcatStrings("FP_ITEM_LESTER_03_0",  
IntToString(Randi)));
```

```
};
```

```
if (Weapon_Gorn_Level_03 > 0)
{
    Randi = Hlp_Random(Weapon_Gorn_Level_03)+1;

    Wld_InsertItem (ItMw_Gorn_04, ConcatStrings("FP_ITEM_GORN_03_0",
IntToString(Randi)));
};

if (Weapon_Diego_Level_03 > 0)
{
    Randi = Hlp_Random(Weapon_Diego_Level_03)+1;

    Wld_InsertItem (ItRw_Diego_04, ConcatStrings("FP_ITEM_DIEGO_03_0",
IntToString(Randi)));
};

if (Weapon_Milten_Level_03 > 0)
{
    Randi = Hlp_Random(Weapon_Milten_Level_03)+1;

    Wld_InsertItem (ItRu_Milten_04, ConcatStrings("FP_ITEM_MILTEN_03_0",
IntToString(Randi)));
};

// Levelpart Four

if (Weapon_Lester_Level_04 > 0)
{
    Randi = Hlp_Random(Weapon_Lester_Level_04)+1;
```

```
        Wld_InsertItem      (ItMw_Lester_05,   ConcatStrings("FP_ITEM_LESTER_04_0",
IntToString(Randi)));
};
```

```
if (Weapon_Gorn_Level_04 > 0)
```

```
{
```

```
    Randi = Hlp_Random(Weapon_Gorn_Level_04)+1;
```

```
        Wld_InsertItem      (ItMw_Gorn_05,    ConcatStrings("FP_ITEM_GORN_04_0",
IntToString(Randi)));
};
```

```
if (Weapon_Diego_Level_04 > 0)
```

```
{
```

```
    Randi = Hlp_Random(Weapon_Diego_Level_04)+1;
```

```
        Wld_InsertItem      (ItRw_Diego_05,   ConcatStrings("FP_ITEM_DIEGO_04_0",
IntToString(Randi)));
};
```

```
if (Weapon_Milten_Level_04 > 0)
```

```
{
```

```
    Randi = Hlp_Random(Weapon_Milten_Level_04)+1;
```

```
        Wld_InsertItem      (ItRu_Milten_05,   ConcatStrings("FP_ITEM_MILTEN_04_0",
IntToString(Randi)));
};
```

```
// Levelpart Five
```

```
if (Weapon_Lester_Level_05 > 0)
```

```
{
```

```
    Randi = Hlp_Random(Weapon_Lester_Level_05)+1;
```

```
    Wld_InsertItem      (ItMw_Lester_06,    ConcatStrings("FP_ITEM_LESTER_05_0",  
IntToString(Randi)));
```

```
};
```

```
if (Weapon_Gorn_Level_05 > 0)
```

```
{
```

```
    Randi = Hlp_Random(Weapon_Gorn_Level_05)+1;
```

```
    Wld_InsertItem      (ItMw_Gorn_06,      ConcatStrings("FP_ITEM_GORN_05_0",  
IntToString(Randi)));
```

```
};
```

```
if (Weapon_Diego_Level_05 > 0)
```

```
{
```

```
    Randi = Hlp_Random(Weapon_Diego_Level_05)+1;
```

```
    Wld_InsertItem      (ItRw_Diego_06,     ConcatStrings("FP_ITEM_DIEGO_05_0",  
IntToString(Randi)));
```

```
};
```

```
if (Weapon_Milten_Level_05 > 0)
```

```
{
```

```
    Randi = Hlp_Random(Weapon_Milten_Level_05)+1;
```

```
    Wld_InsertItem      (ItRu_Milten_06,    ConcatStrings("FP_ITEM_MILTEN_05_0",
```

```
IntToString(Randi));
```

```
};
```

```
// Sekundärwaffen - Stufe 2
```

```
Randi = Hlp_Random(Weapon_Diego_Second_01)+1;
```

```
if (Randi < 10)
```

```
{
```

```
Wld_InsertItem (ItMw_Diego_02, ConcatStrings("FP_ITEM_DIEGO_0",  
IntToString(Randi)));
```

```
}
```

```
else
```

```
{
```

```
Wld_InsertItem (ItMw_Diego_02, ConcatStrings("FP_ITEM_DIEGO_",  
IntToString(Randi)));
```

```
};
```

```
Randi = Hlp_Random(Weapon_Milten_Second_01)+1;
```

```
if (Randi < 10)
```

```
{
```

```
Wld_InsertItem (ItMw_Milten_02, ConcatStrings("FP_ITEM_MILTEN_0",  
IntToString(Randi)));
```

```
}
```

```
else
```

```
{
```

```
Wld_InsertItem (ItMw_Milten_02, ConcatStrings("FP_ITEM_MILTEN_",  
IntToString(Randi)));
```

```
};
```

```
// Sekundärwaffen - Stufe 3
```

```
if (Weapon_Diego_Second_02 > 0)
```

```
{
```

```
    Randi = Hlp_Random(Weapon_Diego_Second_02)+1;
```

```
    if (Randi < 10)
```

```
    {
```

```
        Wld_InsertItem      (ItMw_Diego_03,  
ConcatStrings("FP_ITEM_DIEGO2_0", IntToString(Randi)));
```

```
    }
```

```
    else
```

```
    {
```

```
        Wld_InsertItem      (ItMw_Diego_03,   ConcatStrings("FP_ITEM_DIEGO2_",  
IntToString(Randi)));
```

```
    };
```

```
};
```

```
if (Weapon_Milten_Second_02 > 0)
```

```
{
```

```
    Randi = Hlp_Random(Weapon_Milten_Second_02)+1;
```

```
    if (Randi < 10)
```

```
    {
```

```
        Wld_InsertItem      (ItMw_Milten_03,  
ConcatStrings("FP_ITEM_MILTEN2_0", IntToString(Randi)));
```

```
    }
```

```
    else
```

```

    {
        Wld_InsertItem (ItMw_Milten_03,
ConcatStrings("FP_ITEM_MILTEN2_", IntToString(Randi)));

    };

};

};

```

Wie ihr sehen könnt hab ich die Waffen allesamt ganz simpel benannt :D Das System ist sehr einfach gehalten: In Levelteil 1 findet man die Waffe Stufe 2 (Stufe 1 ist die Ausgangswaffe), in Levelteil 2 die 3 usw. Bei den Sekundärwaffen wird das ausgelost. Stufe 2 wird in Level 1 oder 2 sein, Stufe 3 in 3, 4 oder 5. Das System ist schon für eine vollständige Kampagne (5 Level) einsetzbar, eingestellt wird die Zahl an Spawnpoints wie gewohnt im Kampagnen-Wahlmenü. Für die Primärwaffen sind die Spawnpoints pro Level allerdings auf 9 beschränkt, kann ich aber notfalls auch noch hochsetzen.

Donnerstag, der 30.07.2009

Nachdem mein PC seit heute wieder läuft (wie erwartet Mainboard defekt) gibt's auch ab heute wieder das Entwicklertagebuch.

Zuerst natürlich die ganzen Sachen auf dem PC hier aktualisiert. Wie angekündigt habe ich mich dann mal an die Anzeigen über die Charaktere gekümmert. Angezeigt werden jetzt Name, Status (Fit, Bewusstlos oder Tot), ein/kein Trank und, falls aktiviert, die Punkte. Das ganze wird aber noch mal grundlegend verändert (wie ist noch geheim :p). Jedenfalls sollte das, wenn es letztlich so aussieht, wie ich mir das vorstelle, viel besser wirken, als jetzt, und vor allem nicht so viel Platz einnehmen.

Code-Beispiel #1

```

// Anzeigen für die Charaktere

// Lester

if ((Npc_IsDead(Char_Einhand))
&& (Mod_Charakter != 1))

|| ((Npc_IsDead(hero))

```

```
&& (Mod_Charakter == 1))  
  
{  
  
    ShowLester = "Lester: Tot";  
  
}  
  
else if ((NPC_IsInState(Char_Einhand, ZS_Unconscious))  
&& (Mod_Charakter != 1))  
  
|| ((NPC_IsInState(hero, ZS_Unconscious))  
&& (Mod_Charakter == 1))  
  
{  
  
    ShowLester = "Lester: Bewusstlos";  
  
}  
  
else  
  
{  
  
    ShowLester = "Lester: Fit";  
  
};  
  
if ((NPC_HasItems(Char_Einhand, ItPo_Health) == 1)  
&& (Mod_Charakter != 1))  
  
|| ((NPC_HasItems(hero, ItPo_Health) == 1)  
&& (Mod_Charakter == 1))  
  
{  
  
    ShowLester = ConcatStrings(ShowLester, ", ein Heiltrank");  
  
}  
  
else  
  
{  
  
    ShowLester = ConcatStrings(ShowLester, ", kein Heiltrank");  
  
};
```


(...)

```
if (Mod_Punkte == 1)
{
    ShowLester = ConcatStrings(ShowLester, ", Punkte: ");
    ShowLester = ConcatStrings(ShowLester, IntToString(LesterPoints));

    (...)
};
```

Sonntag, der 02.08.2009

Heute habe ich mal wieder was gemacht. Und zwar habe ich mit Milten angefangen und ihm neben seinem Feuerpfeil einen "Skelett beschwören" Zauber gegeben. Das Skelett folgt Milten und kämpft für diesen mit. Ist aber noch nicht gebalanced und noch nicht ausgiebig getestet, nur ein bisschen ansatzweise. Wem also was auffällt: Ihr wisst ja, wo ihr mich findet :p

Mittwoch, der 19.08.2009

Heute ging's endlich mal weiter, wenn auch nur mit einer Mini-Änderung. Und zwar werden jetzt die Kills des Skelettes von Milten diesem auf sein Punktekonto gutgeschrieben.

In den nächsten Tagen wird's wieder mehr geben, hoffe ich, hab nur zur Zeit viel für XR zu tun und das hat Vorrang ;)

Donnerstag, der 20.08.2009

Heute habe ich mich um den Dialog mit Milten gekümmert, wo man das Beschwören des Skelettes abstellen kann (und natürlich auch wieder anstellen). Das ganze wird dann einfach über eine Variable geregelt.

Ob ich morgen was mache, weiß ich noch nicht, am Samstag auf jedenfall nicht, da bin ich auf der GamesCom, also keine Zeit :)

Montag, der 21.09.2009

Heute ging's hier endlich mal wieder ein bisschen weiter.

Und zwar habe ich eingebaut, dass die Begleiter an beliebigen Waypoints warten können. Ob der momentan integrierte „Warte hier.“ – Dialog enthalten bleibt, weiß ich noch nicht, hauptsächlich ist das Feature für spezielle Orte in der Kampagne gedacht, wo man alleine oder zu zweit oder so unterwegs ist. Da ich das System allgemein gehalten habe, habe ich einen TA für jeden beliebigen Waypoint im Level. Davon merkt der Spieler nicht viel, ist aber für mich ein großes Zeitersparnis (vor allem, wenn man die Begleiter überall warten lassen kann).

Sonntag, der 11.10.2009

Heute habe ich zuerst mal die Bezeichnung des Testlevel im Dialogmenü geändert. So heißt das nun Testlevel und nicht mehr Kampagne 1, da ja irgendwann wirklich die erste Kampagne kommt :)

Dann habe ich einen kleinen Bugfix vorgenommen. Man konnte bei Milten gar nicht mehr das Beschwören des Skelettes abstellen.

Und zum Abschluss des Tages gab's von ThielHater dann einen neuen Font. Damit gibt's noch Probleme, deswegen gab's auch kein Update im Forum.

Montag, der 12.10.2009

Heute ging's dann mit einer neuen Einstellung im Punkteanzeigen-Menü weiter. Man kann dort nun zwischen vier Optionen wählen. Zum einen kann man nach wie vor die Punkte entweder anzeigen lassen oder eben nicht, zum Anderen kann man nun noch die Reihenfolge der Namen entweder statisch anzeigen lassen, d.h. die bleiben, wie sie sind (alphabetische Reihenfolge), oder man wählt die dynamische Anzeige, dann hängt die Reihenfolge von den Punkten ab (absteigend sortiert, höchsten Punkte oben, bei

Gleichstand alphabetisch).

Code-Beispiel #1

// Dynamische Rangliste

```
if (Mod_Punkte == 1)  
|| (Mod_Punkte == 2)  
{  
  
    if (DiegoPoints >= GornPoints)                                // Diego Platz 1  
        && (DiegoPoints >= LesterPoints)  
        && (DiegoPoints >= MiltenPoints)  
    {  
  
        DiegoPos = Pos1;  
  
  
        if (GornPoints >= LesterPoints)  
            && (GornPoints >= MiltenPoints)  
        {  
  
            GornPos = Pos2;  
  
  
            if (LesterPoints >= MiltenPoints)  
            {  
  
                LesterPos = Pos3;  
  
                MiltenPos = Pos4;  
  
            }  
  
        else  
  
        {  
  
            MiltenPos = Pos3;  
  
            LesterPos = Pos4;
```

```
};  
  
}  
  
else if (GornPoints >= LesterPoints)  
&& (GornPoints < MiltenPoints)  
{  
  
    MiltenPos = Pos2;  
  
    GornPos = Pos3;  
  
    LesterPos = Pos4;  
  
}  
  
else if (GornPoints >= MiltenPoints)  
&& (GornPoints < MiltenPoints)  
{  
  
    LesterPos = Pos2;  
  
    GornPos = Pos3;  
  
    MiltenPos = Pos4;  
  
}  
  
else  
{  
  
    GornPos = Pos4;  
  
    if (LesterPoints >= MiltenPoints)  
    {  
  
        LesterPos = Pos2;  
  
        MiltenPos = Pos3;  
  
    }  
  
    else  
    {  
  
        MiltenPos = Pos2;
```

LesterPos = Pos3;

};

};

}

(...)

Jeder hat eben eine Position, z.b. DiegoPos, dort wird der Platz gespeichert. Die Variable wird dann beim Printen angegeben. Pos1, Pos2, Pos3 und Pos4 sind die Konstanten, die die Höhe speichern.

Von ThielHater gab's nun nochmal eine neue Version des Fonts, das klappt nun fast ganz korrekt, werden wir wohl morgen nochmal angehen.

Und kurz vor dem Upload der neuen Version 0.3.7 hab ich noch ein Problem mit dem Skelett behoben, welches viel zu früh angegriffen hat (verglichen mit den anderen Nahkämpfern).

Montag, der 02.11.2009

Heute hat Jim eine leicht überarbeitete Version des Hauptmenüstückes fertig gestellt.

Freitag, der 20.11.2009

Aufgrund der vielen Beiträge im L4G-Thread und der Meldung habe ich heute wieder was gemacht, wenn es auch nur ein kleiner Fix war. So hat das beschworene Skelett nun keine Fokus mehr.

Samstag, der 21.11.2009

Heute wurde erstmal das "Gegnern" bei der Anzeige der übrigen Gegner in "Infizierten" geändert.

Dann habe ich noch eine weitere Sieg-Meldung eingebaut, nämlich dann, wenn man das Ende der Map erreicht.

Und nun habe ich noch L4G dem neuen Teutonic-Standard angepasst. So kommt jetzt auch hier eine Sicherheits-Abfrage, wenn man ein neues Spiel startet, ob man das wirklich will.

Da Edain weiterhin auf den Fehler des Skelettes bestanden hat, hab ich mal ein wenig gesucht und getestet und letztlich den Fehler gefunden. Da hatte sich doch tatsächlich ein Abfrage im Kampf-State versteckt ^^

Sonntag, der 25.04.2010

Nicht heute eingebaut, aber da ich erst heute hier weiter schreibe halt unter dem heutigen Datum. Die letzten Wochen habe ich wieder ein paar Kleinigkeiten geschafft (mit Unterstützung).

Zum einen gibt es jetzt eine optimierte Anzeige der Icons links oben. Die wurden in der letzten Version noch teilweise an den Seiten abgeschnitten, das passiert nun nicht mehr (Textur von ThielHater, Tipp dazu zum Abschneiden von NicoDe). Das ganze lag an der zu großen Trennlinie zwischen den einzelnen Zeichen (darf nur einen Pixel breit sein, waren aber 4 oder so).

Dann habe ich auf den Wunsch von Helljumper nun alle Zombies in Infizierte umgetauft und hoffentlich nichts vergessen (Stampfer hilft, ist ja seine einzige Verwendung bei mir, das Suchen in Dateien [na gut, ich schau auch immer mal die nächste freie ID nach]).

Und dann gibt's noch das „Todeskammer“ - Prinzip aus dem MP von Left 4 Dead. „Stirbt“ einer der vier Gefährten, so ist er nicht gleich tot, sondern landet in dieser Kammer. Man sollte ihn nun befreien, um ihn wieder mitnehmen zu können (je weniger man dabei hat, desto schwieriger, hoffe ich zumindest ^^). Der Held landet natürlich auch dort und wird dann von der KI befreit. Da fehlt noch ein Respawn von Infizierten auf dem Weg zurück (wenn man also schon nach der „Todeskammer“ war und dann zurück will zum befreien müssen da noch Infizierte hin, damit's schwieriger wird). Sind alle vier Gefährten in der Kammer, dann ist das Spiel vorbei. Da gibt's jetzt übrigens auch eine neue Meldung sowie ein automatisches Ende, ebenso beim Sieg.

Und zu guter Letzt gab's noch eine Erweiterung der Punkte: Die Bosse geben jetzt unterschiedlich viele Punkte (Tank: 5, Witch: 4, Hunter/Boomer: 3), Aufhelfen, Befreien und Trank geben werden nun ebenfalls belohnt.

Das waren dann auch alle Änderungen für die Version 0.4.0, werde sehen, dass jetzt wieder öfter was kommt, diese Woche werden hoffentlich auch die Story-Planungen starten.

Montag, der 26.04.2010

Da gestern und heute ja ziemlich viel Feedback kam, hab ich heute gleich noch einige Optimierungen vorgenommen.

Im Hauptmenü kam neben dem neuen Stück auch noch das alte G2-Stück. Das ist nun nicht mehr so (hoffe ich :D). Das L4G-Stück ist noch ein wenig leise, das wird sicherlich irgendwann mal geändert, das liegt am Direct Music Producer, der macht die Stücke immer ein ganzes Stück leiser bei der Umwandlung.

Dann habe ich heute den Tank gefixt, der greift nun richtig an und ist dadurch nicht zu unterschätzen (ich sage nur vier Schläge :p).

Der Boomer wurde dafür ein wenig geschwächt, lockt nun aber, wenn er in der Nähe eines Überlebenden stirbt, pro Überlebenden in der Nähe zehn neue Infizierte an. Das ist ziemlich fies, wie ich finde.

Monat, der 05.07.2010

Heute ging's nach einer weiteren längeren Pause wieder mal weiter.

Nun verschwindet nach dem Einsammeln einer Waffe auf dem leichten Schwierigkeitsgrad der Effekt aus der Welt.

Und als tollere Neuerung gibt's dann nun den ersten neuen Boss seit langem. Dabei handelt es sich um den Hordenführer. Dieser spawnt alle zwei Sekunden (mal sehen, wie sich das auswirkt, fünf Sekunden waren jedenfalls langweilig) einen neuen Infizierten. Seine Werte sind relativ schlecht und er gibt auch nur drei Punkte, könnte aber durch das Spawnen trotzdem ziemlich gefährlich werden.

Dienstag, der 06.07.2010

Heute habe ich als neues Feature eingebaut, dass die Tageszeit nun ausgewürfelt wird bei Levelstart. So sollte es nun bei jedem Spielstart eine andere Tageszeit geben. Auch ob es regnet oder nicht wird zu Beginn ausgewürfelt. Ich kann natürlich immer noch für spezielle Level manuell einstellen, wie das Wetter/die Uhrzeit da sein soll.

Mittwoch, der 07.07.2010

Heute habe ich mit dem Spitter angefangen. Nach einigen kleineren Problemen durch das Abschalten der Magie wollte der erst nicht funktionieren, jetzt klappt er aber und ist wie der der Hordenführer auch am besten in Kombination mit normalen Infizierten.

Dann gab es noch eine Meldung mit dem aktuellen Monsternamen, die ich jetzt auch entfernt habe.

Donnerstag, der 16.06.2011

Heute ging es nach einer sehr lange Pause wieder etwas voran.

Eingebaut habe ich unsere Online-Features. Wahlweise kann man sich jetzt L4G mit seinem Account bei uns verknüpfen, den man schon für die GoMoN-Preview oder Elemental War erstellt hat und bekommt dort dann zusätzliche Anzeigen. Derzeit werden neben den Highscores auch die Spielzeit und die gespielten Charaktere übertragen.

Freitag, der 17.06.2011

Heute gab's ein paar Fixes bei den Punkten, es war möglich, dass die NPCs ewig viele Punkte bekommen haben, weil die Wiederbelebung mehrfach gezählt wurde.

Sonntag, der 26.06.2011

Heute habe ich noch eine Kleinigkeit für die Niederlage behoben, das wird nun gleich erkannt, wenn alle gefangen oder KO sind, es müssen also nicht alle schon im „Gefängnis“ sein.

Dann habe ich noch die Möglichkeit für die alte CPU rausgenommen, da die Schleife jetzt wichtiger ist und ein Frametrigger, also auch gar nicht mehr so oft läuft wie vorher.

Experimentalerweise habe ich auch mal die Manabar verschoben und sie wird jetzt dauerhaft angezeigt. Der Code dazu stammt für das verschieben letztlich von Milky-Way (zumindest die Grundidee) und das dauerhafte Anzeigen von Gottfried, Lehona und NicoDe.

Donnerstag, der 30.06.2011

Weil ja mal wieder das Problem mit den bösen im Weg stehenden Nahkämpfern aufkam, habe ich dafür mal eine Funktion geschrieben, die deren Entfernung zur Schussbahn ermittelt und wenn die zu nah sind, die Nahkämpfer strafen lässt. Das ganze funktioniert bisher nur für die NPCs Gorn und Lester (Nahkampf) und Diego und Milten (Fernkampf). Wenn das System so gut klappt, dann wird das noch um den Held und die restlichen Optionen erweitert.

Dann habe ich noch das Problem behoben, dass manchmal die Welt-Waffen nicht ausgetauscht wurden und somit der Effekt weiterhin sichtbar war.