LRES – Das Lightweight Roleplaying Experience System

1. Über das LRES

Das LRES ist ein System das einem Spieler Erfahrungspunkte nach seiner Rollenspielaktivität zuteilt. Dabei wird Rollenspielaktivität über die Anzahl Gespräche/Emotes ermittelt, im Zusammenhang damit wie viele andere Spieler daran teilhaben. Ziel des Systems ist es nicht Charakteren Erfahrungspunkte zu geben, die alleine vor sich hinspielen und emoten/sprechen. Eines der Hauptziele ist es ein möglichst einfach zu installierendes System zu schaffen das sich möglichst problemlos in vorhandene Module und persistene Welten einbauen läßt. Ein weiteres Ziel ist es die Ressourcen des Servers im Heartbeat nicht zu sehr zu belasten. Außerdem soll das System wärend der Laufzeit des Moduls in allen wichtigen Dingen einstellbar sein.

2. Funktionsweise

Jeder Spieler verfügt über ein Dialogpunktekonto. Für jeden anderen Spieler (oder entsprechend eingestelltes Objekt) das einen Text von diesem Spieler höhren kann, erhält er einen Punkt auf seinem Konto gutgeschrieben, sofern der Dialog gewisse Anforderungen an Länge und nichtvorhanden sein von OOC Flags erfüllt. Wenn sein Punktekonto einen gewissen Betrag übersteigt und seine letzte EP Vergabe (oder der Login) einen bestimmten Zeitraum zurückliegt, bekommt er aus einem Intervall zufällig ausgewählte EP gutgeschrieben. Dabei werden alle Dialogpunkte bis auf ein Residuum gelöscht und das Konto muß wieder aufgefüllt werden um wieder für eine Vergabe in Frage zu kommen. Ein Spieler kann von der SL auch für die Vergabe von Punkten geblockt werden, so das er keine weiteren EP über das Skript erhält. Wenn der Spieler dann wieder freigeschaltet wird, werden seine Werte so resetted, als hätte er gerade EP bekommen. Er startet also bis auf sein Residuum von vorne. Für jeden Sieler der einloggt wird die minimale Anzahl an Dialogpunkten die nötig ist um einen einstellbaren Wert erhöht. Alle diese Werte sind wärend des Spiels mit dem SL Stab anpassbar. Außerdem kann das LRES so geschaltet werden, das jeder Spieler einen zufälligen Zeitraum, aus einem Zeitintervall, zwischen zwie EP Vergaben warten muß. Des weiteren kann man einen Linearfaktor für die zu vergebenden EP einschalten, der bei (zufällige Wartezeit)/(minimale Wartezeit) liegt, so das Spieler die länger warten müssen auch mehr EP bekommen.

3. Einstellungen mittels des Stabes

Wenn der Stab von einem SL auf sich selber angewendet wird, kommt man in das normale Einstellungsmenü:

3.1: Der minimale EP Wert:
Dieser Wert ist die Untergrenze an EP die ein Charakter bei einer Vergabe bekommen kann.

3.2: Der maximale EP Wert:
Dieser Wert ist die Obergrenze an EP die ein Charakter bei einer Vergabe bekommen kann.

3.3: Minimale Zeit zwischen zwei Vergaben:
Dies ist die minimale Zeit die zwischen der letzen EP Vergabe, oder dem einloggen, und einer weiteren Vergabe mindestens vergangen sein muß.

3.4: Maximale Zeit zwischen zwei Vergaben:
Die ist die maximale Zeit die zwischen zwei EP Vergaben gewartet werden muß (natürlich bekommt man nur EP wenn man auch genügend Dialogpunkte hat).

3.5: Der maximal übernommene Wert an Dialogpunkten (Residuum)
Dies ist der maximale Wert an Dialogpunkten der nach einer EP Vergabe übrig bleibt.

3.6: Alle wieviel Minuten geprüft werden soll
Dies gibt die Granularität der EP Vergaben an. Alle eingestellte Anzahl an Minuten prüft das Skript ob ein Spieler für eine EP Vergabe in frage kommt und vergibt im positiven Falle die EP. Diese Zahl ist fast unabhängig von der minimalen Zeit zwischen 2 Vergaben, sollte aber deutlich niedriger liegen, da sonst die Vergabe zu grobkörnig wird. Alle 2-3 Minuten sollte eigentlich genügen.

3.7: Die minimalen Dialogpunkte für eine Vergabe:
Dieser Wert gibt die minimale Anzahl an Dialogpunkten an, die ein Spieler braucht um für eine EP Vergabe in Frage zu kommen. In diesem Wert ist bereits die Erhöhung für die Anzahl der eingeloggten Spieler enthalten. Nach dem Ändern werden keine Punkte mehr für die eingeloggten Spieler aufaddiert. Der Wert ist also absolut.

3.8: Erhöhung der Dialogpunkte pro Spieler:
Dieser Wert gibt an um wie viel sich die minimal nötige Anzahl an Dialogpunkten pro eingeloggtem Spieler erhöht.

3.9: Zurücksetzen auf Defaultwerte:
Alle veränderlichen Werte des LRES werden auf die in mvd_02_const definierten Werte zurückgesetzt. Dabei wird die Erhöhung der Dialogpunkte pro Spieler angewendet. Der zu erreichende Wert ist nach einem reset also der in mvd_02_const angegebene Minimalwert +(Anzahl Spieler)* (Dialogpunkteerhöhung pro Spieler).

3.10: Die zufällige Minimalzeit zwischen zwei Vergaben an-/ausschalten:
Schaltet die zufällige Wartezeit zwischen zwei Vergaben an bzw. aus.
3.11: Den Linearfaktor ein-/ausschalten:
Schaltet den Linearfaktor für die EP Vergabe an bzw. aus.
3.12: Anzeigen der momentanen Werte:
Zeigt die momentane Einstellung des LRES an.
3.13: Lauf erzwingen:
Dies erzwingt einen Überprüfungslauf, in dem geprüft wird, ob ein Spieler EP bekommen soll, und diese dann auch verteilt.
3.14: Das LRES aktivieren/deaktivieren:
Aktiviert. bzw deaktiviert das LRES.

Wird der Stab hingegen auf einen Spieler angewendet, so zeigt er die aktuellen LRES Werte des Spielers an, und man hat folgende Auswahlmöglichkeiten:

3.15: Den Spieler sperren:
Der Spieler ist für die Zukunft von allen EP Vergaben durch das LRES ausgeschlossen.
3.16: Den Spieler entsperren:
Hebt eine Sperre wieder auf und resetted die Werte des Spieler. Es bleibt also nur sein Residuum an Dialogpunkten übrig. Alle anderen wärend der Sperre erlangten Dialogpunkte verfallen. Ebenso muß er wieder die für ihn gültige Minimalzeit warten bis zur nächsten Vergabe.
3.17: Den Spieler resetten:
Resetted den Spieler wie unter 3.16 angegeben, aber ohne seinen gesperrt/entsperrten Zustand zu beeinflussen.

4. Installation

Zur Installation muß das beigelegte ERF eingebunden werden. Das Placeable „LRES Main Hub“ muß irgendwo im Modul plaziert werden. Der Einstellungsstab ist optional, wird aber gebraucht wenn man das LRES wärend der Laufzeit konfigurieren will, oder einen Spieler blockieren/freigeben will. Der beste Platz für den „LRES Main Hub“ ist eine Sl Area die für Spieler nicht zugänglich ist. Zwar ist der Einstellungsstab nicht von Spielern benutzbar, er sollte aber auch nur an SL vergeben werden. Der Stab befindet sich unter Selbsterstellte Gegenstände->Spezial->Spezial1. Der LRES Main Hub befindet sich unter Selbsterstellte Placeables->Spezial->Spezial1. Weiterhin müssen in den Events bzw. Skripten entsprechend der folgenden Tabelle Funktionen eingebunden werden:

Objekt	Event/Skript	Include	Funktion
Modul	OnHeartBeat	mvd_02_modheart	MvD_02_ModuleHeartBeat()
Modul	OnClientEnter	mvd_02_init	MvD_02_PlayerEnter()
Modul	OnClientLeave	mvd_02_init	MvD_02_PlayerExit()
Modul	OnModuleLoad	mvd_02_init	MvD_02_ModInit()
Skript	nw_g0_conversat	mvd_02_conversat	MvD_02_Conversat()

Es wird empfohlen die Skripte mvd_02_userevent, mvd_02_giver, mvd_02_conversat, nw_g0_conversat und nw_ch_ac4 (siehe auch den NWN Familiar Bug Workaround) zu cachen.

Soll ein NSC oder ein anderes Objekt ebenfalls Dialogpunkte verteilen, wie ein anderer Spieler es tun würde, so müssen die Events durch die Aufrufe folgender Funktionen abgeändert werden:

Event	Include	Funktion
OnSpawn	mvd_02_init	MvD_02_NSCInit(OBJECT_SELF)
OnConversation	mvd_02_conversat	MvD_02_ConversatNPC()

Es wird nicht empfohlen solche Objekte aufzusetzen, da sie von den Spielern leicht mißbraucht werden können.

4.1: Installation des NWN Familiar Bug Workarounds

Der NWN Familiar Bug sorgt dafür das wenn der Meister eines Familiars seinen eigenen Familiar sprechen hört, wenn er von ihm Besitz ergrifen hat, läßt sich automatisch die Besitzergreifung und der Fokus geht zurück auf den Meister. Dies ist ein Problem von NWN und nicht des LRES.
Um den Workaround für diesen Bug zu installieren, ist folgendes zu tun:

1. Den Aufruf von Summon Familiar() in nw_s2_familiar durch MvD_02_SummonFamiliar(OBJECT_SELF) zu ersetzen. Es muß dabei per include Befehl die Datei mvd_02_familiar eingebunden werden.
2. Jeden anderen Aufruf von SummonFamiliar(oMaster) in nicht NWN Skripten durch MvD_02_SummonFamiliar (oMaster) zu ersetzen. (oMaster wird hier als Beispiel genutzt die Parameter für MvD_02_SummonFamiliar müssen die selben sein, wie für den SummonFamiliar Aufruf). Es muß dabei per include die Datei mvd_02_familiar eingebunden werden.
3. In er Datei nw_ch_ac4 Muß am Ende ein Aufruf von MvD_02_ConversatFamiliar() erfolgen. Dabei muß die Datei mvd_02_conversat per include eingebunden werden.

4.2: Änderungen an den OOC Tests und dem OOC Area Marker

In dem Skript mvd_02_conversat sorgen die beiden Funktionen MvD_02_OOCMatchCase und MvD_02_OOCMatchNoCase für den Abgleich mit den OOC Strings aus mvd_02_const. MvD_02_OOCMatchCase vergleicht den gesprochenen String mit allen Konstanten der Form cMvD_02_sOOC[Nummer]. Dieser Abgleich ist sensitiv was Groß- und Keinschreibung angeht. D.h. „Test“ ist ungleich „test“. Die Funktion MvD_02_OOCMatchNoCase vergleicht den gesprochenen String mit allen Konstanten der Form cMvD_02_sOOCWord[Nummer]. Dieser Vergleich ist nicht sensitiv was Groß- und Kleinschreibung angeht, allerdings müssen die OOC-Word Konstanten in kleinschreibweise vorliegen. Um weitere Tests hinzuzufügen reicht es eine weitere Konstante der gewünschten Art in mvd_02_const auszuweisen und dann eine weitere else-if Klausel in der entsprechenden Funktion einzubauen. Sollen Tests herausgenommen werden, so reicht es die entsprechende else-if Klausel auszukommentieren. Sollen einfach nur andere Strings benutzt werden, so reicht es die Werte der Konstanten umzusetzen.
Jede Area, deren Name den substring cMvD_02_sOOCAreaName enthält ist als OOC Area gekennzeichnet. In diesen Areas werden keine Dialogpunkte vergeben. Um einen eingenen Kennzeichner zu benutzen muß man einfach cMvD_02_sOOCAreaName abändern.

4.3: Wichtige Konstanten in mvd_02_const

Name	Default	Beschreibung
cMvD_02_iListenPatternNumber	5000	Die Standartnummer des ListenPatterns
cMvD_02_iPCRandomTime	TRUE	Gibt an ob jeder Spieler eine unterschiedliche Randomtime haben soll
cMvD_02_iStrMinLength	2	Die minimale Länge die ein String haben muß um zu zählen
cMvD_02_iCountPerPC	5	Der Wert um den die minimal Gialogpunkteanzahl pro Spieler erhöht werden
cMvD_02_iMinDialogCount	25	Die minimale Anzahl an Dialogpunkten die man braucht für 0 Spieler
cMvD_02_iHeartBeatTime	2	Alle wie viele Minuten das Skript prüfen soll, ob EP Vergaben werden müssen.
cMvD_02_iMinTimeDiff	7	Der minimale Minutenwert nachdem man wieer EP bekommen kann. Dieser Wert wird benutzt falls keine Zufallszeit pro Spieler eingestellt ist.
cMvD_02_iMaxTimeDiff	15	Der maximale zufällige Zeitwert pro Spieler
cMvD_02_iMaxResiduum	10	Wie viele Dialogpunkte nach einer EP Vergabe oder reset übrig bleiben
cMvD_02_iMinXP	10	Der minimale EP Wert der vergeben wird
cMvD_02_iMaxXP	60	Der maximale EP Wert der vergeben wird
cMvD_02_iIsXPMult	FALSE	Ob ein linearer Zuschlagssatz benutzt werden soll
cMvD_02_sOOCX		Ein Substring der einen Text als OOC ausweist. Case sensitiv
cMvD_02_sOOCWordX		Ein Substring der einen Text als OOC ausweist. Case insensitiv
cMvD_02_sHubTag	LRESMainHub	Diese Konstante definiert den Tag des LRES Main Hubs, damit ihm ein UserDefinedEvent gesendet werden kann
cMvD_02_UserEventNr	5000	Nummer des UserdefinedEvents der dem Hub gesendet wird.
cMvD_02_sOOCAreaName	OOC	In einer Area die diesen String (case sensitiv) in ihrem Namen trägt, werden keine Dialogpunkte vergeben. OOC Area

5. Beispielmodul

Enthalten ist ein Beispielmodul mit einem NSC der genau wie sonst ein anderer Spieler auf die Dialog der Spieler hört. Außerdem gibt es eine weitere für Spieler unzugänglichen Area, in der der Einstellungsstab liegt und sich der LRES Main Hub befindet.Die Skripte die nicht mit mvd_02 anfangen sind die nach dieser Anleitung erstellten bzw. modifizierten Skripte. Alle Skripte die nicht mit einem Header von mir versehen sind, sind modifizierte Versionen der original Bioware Skripte. Alle Modifikationen oder Additionen in den Originalskripten können durch eine Suche nach „MvD-“ gefunden werden (sie sind durch Kommentare gekennzeichnet).

Benutzung: Vor den NSC stellen und ein paar Dialoge sprechen (32 Texte sollten ausreichen. Bitte beachten, das es einige Marker gibt die einen Text als OOC ausweisen. Solche Texte werden nicht als Dialog angerechnet. Des weiteren erhöht sich die nötoge Dialoganzahl pro eingeloggten Spieler um 5 [für einen Spieler ist es schon in der 32 eingerechnet]). Dann einfach zwischen 7 und 15 Minuten warten, dann sollte der Spieler EP bekommen. Es kann sein das man etwa 15 Sek warten muß, bis der NSC Dialoge annimmt. Dies liegt daran, das das OnSpawn manchmal erst etwas spät gestartet wird. Dies wird von NWN bestimmt und nicht vom LRES.

6. Version und geplante Erweiterungen

LRES-Version: LRES-0.9-Beta-9.9
Anleitung Version: 0.9.7

7. Copyright

Das LRES, sowohl das ERF File, wie auch der gesammte Inhalt des LRES Packetes, des Moduls und diese Anleitung ist ©2004,2005 MvDunkelfels.

8. Haftungsausschluß

Dieses Packet wird unter Ausschluß jeglicher Haftung des Autors zur Verfügung gestellt. Der Autor haftet für keinerlei Schäden, Unannehmlichkeiten oder jedwede Form von Nachteilen die dem benutzer mittelbar oder unmittelbar durch Benutzung, Betrachtung oder sonstige Verwendung entstehen.

9. Kontakt

Web: http://de.geocities.com/mvd_moragon/
E-Mail: mvd_moragon@yahoo.de

Bugreports bitte an die oben genannte Adresse.
Spam, Flames etc. werden nach /dev/null geschickt.