

Indice degli Argomenti delle Demo

| N° | Argomento della Demo in Ordine Alfabetico |
|----|--|
| 1 | Download |
| 2 | Raccolta Campioni |
| 3 | Animations with TRNG - Animazioni con TRNG |
| 4 | Boats - Barche |
| 5 | Cameras Demo - Demo Telecamere |
| 6 | Chronicles Objects - Oggetti Chronicles |
| 7 | Collisions under TRNG - Collisione Sotto TRNG |
| 8 | Diary and Images - Diario e Immagini |
| 9 | Elevators and Detectors - Ascensori e Rilevatori |
| 10 | Fish and Pirahnas - Pesci e Pirania |
| 11 | FMVs: Video Cutscenes - Filmati Video e Cutscene |
| 12 | Fog Distance tutorial - Nebbia a Distanza Tutorial |
| 13 | Harpoon and Frog-Man - Arpione e Uomo Sub |
| 14 | Input Box and StaticMip - Casella di input e StaticMip |
| 15 | Jeep and Side-Car - Jeep e Sidecar |
| 16 | Kayak - Kayak |
| 17 | Lightnings and Cutscenes - Lampi e Cutscene |
| 18 | Mirrors - Specchi |
| 19 | Miscellaneous - Miscellaneous |
| 20 | Miscellaneous II - Miscellaneous II |
| 21 | New Font - Nuovo Tipo di Carattere |
| 22 | New Mirrors - Nuovi Specchi |
| 23 | Parallel-Bars - Barre Parallel |
| 24 | Plugin Demo - Demo Plugins |
| 25 | Quick-Sands - Sabbie Mobili |
| 26 | SavegamePanel - Nuovo Pannello Salva Gioco |
| 27 | Stand-by Mode - Modalità standby |
| 28 | Statics Demo - Demo Static |
| 29 | TRNG Variables - Variabili TRNG |
| 30 | Vertical Triggers - Triggers Verticali |

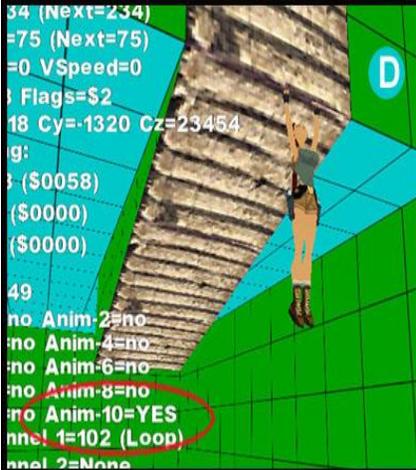
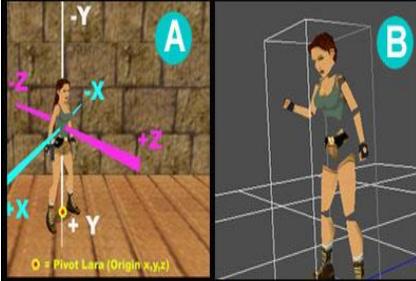
Download nelle Demo by Paolone

| N° | Download | Argomento della Demo |
|-----|--|--|
| 1° | AnimationTutorial.zip (4 Mb) | Animazioni con TRNG (Animations with TRNG) |
| 2° | Boats_new.zip (1.78 MB) | Barche (Boats) |
| 3° | Cameras.zip (101 Kb) | Demo Telecamere (Cameras Demo) |
| 4° | Chronicles.zip (3.8 MB) | Oggetti Chronicles (Chronicles Objects) |
| 5° | Collisions.zip (3.4 Mb) | Collisione Sotto TRNG (Collisions under TRNG) |
| 6° | Diary_Images.zip (5.9 MB) | Diario e Immagini (Diary and Images) |
| 7° | Elevator_Detectors.zip (5.98 MB) | Ascensori e Rilevatori (Elevators and Detectors) |
| 8° | Fish_and_Pirahnas.zip (3 MB) | Pesci e Pirania (Fish and Pirahnas) |
| 9° | video_fmV_cutscenes.zip (22 MB) | Filmati Video e Cutscene (FMVs: Video Cutscenes) |
| 10° | Fog_Distance_Tutorial.zip (381 Kb) | Tutorial Nebbia a Distanza (Fog Distance tutorial) |
| 11° | Harpoon.zip (2.4 Mb) | Arpione e Uomo Sub (Harpoon and Frog-Man) |
| 12° | InputBox_StaticMip (4.4 Mb) | Casella di input e StaticMip (Input Box and StaticMip) |
| 13° | JeepBike.zip (2.1 MB) | Jeep e Sidecar (Jeep and Side-Car) |
| 14° | Kayak.zip (3.6 MB) | Kayak (Kayak) |
| 15° | (Download mancante) | Lampi e Cutscene (Lightnings and Cutscenes) |
| 16° | Mirrors.zip (1.04 Mb) | Specchi (Mirrors) |
| 17° | Mis_Sample_1.zip (2.9 MB) | Miscellaneous (Miscellaneous) |
| 18° | Miscellaneous II (4 MB) | Miscellaneous II (Miscellaneous II) |
| 19° | NewFont.zip (2.5 Mb) | Nuovo Tipo di Carattere (New Font) |
| 20° | New Mirrors (3.19 MB) | Nuovi Specchi (New Mirrors) |
| 21° | Parallel_Bars.zip (2.2 Mb) | Barre Parallel (Parallel-Bars) |
| 22° | Plugin_Obj_Demo.zip (3.9 MB) | Demo Plugins (Plugin Demo) |
| 23° | QuickSand.zip (2.5 Mb) | Sabbie Mobili (Quick-Sands) |
| 24° | Savegame_Panel.zip (8.8 MB) | Nuovo Pannello Salva Gioco (Savegame Panel) |
| 25° | Standby.zip (3 MB) | Modalità standby (Stand-by Mode) |
| 26° | Statics.zip (2.4 Mb) | Demo Static (Statics Demo) |
| 27° | TRNG_Variables.zip (3.2 Mb) | Variabili TRNG (TRNG Variables) |
| 28° | Vertical_Triggers.zip (11 Kb) | Triggers Verticali (Vertical Triggers) |

Demo by Paolone traduzione in Italiano by Franky

| Il mio Adorato Leo | Raccolta dei Campioni |
|---|--|
|  | <p>In questa tabella è possibile trovare tutte le Demo create per mostrare le nuove funzionalità di prossima generazione.</p> |
| | <p>L'elenco è in ordine alfabetico, in modo da poter scorrere per individuare la Demo desiderata, utilizzando le funzioni di ricerca nel Browser.</p> |
| | <p>Suggerisco, non solo di giocare questi mini-livelli, ma anche di studiare attentamente i files di origine, come lo Script.txt e il file di progetto di livello (.PRJ).</p> |
| | <p>Questo è il modo più semplice per comprendere alcune caratteristiche complesse del motore TRNG.</p> |
| | <p>NOTA: Tutte le Demo sono state sviluppate solo in formato sorgente (.PRJ e Script.txt), per questo motivo è necessario costruire lo Script.txt, Script.txt copia e English.txt nella cartella Script e quindi avviare NG_Center e fare clic su [pulsante Built].</p> |
| | <p>Stesso discorso vale per i files .PRJ, caricare il progetto nel programma di NGLE e fare clic sul pulsante [Exit & Play].</p> |
| | <p>Se la Demo contiene due o più progetti è meglio utilizzare NGLE solo al fine di eseguire un'uscita Wad, e quando si è creato il file .Tom per tutti i progetti, è possibile utilizzare NG_Tom2Pc.exe per caricare e costruire tutti i file .Tom della demo nel momento stesso.</p> |
| <p>Si prega di leggere sempre il file Read_Me.txt che si trova in tutte le Demo, alcune possono funzionare solo da una determinata versione di TRNG in poi.</p> | |
| <p>FINE</p> | |

ScreenShots



Descrizione

Animazioni con TRNG (Animations With TRNG)

[Download: AnimationTutorial.zip \(4 Mb\)](#)

Questo file unisce un progetto di un livello con un Tutorial [Html](#) sulla creazione di animazioni personalizzate con il motore TRNG.

Potete trovare il tutorial in un altro file zip denominato "[Help_Animation_Command.zip](#)", all'interno del file "[AnimationTutorial.zip](#)".

A) Nel tutorial ci sono molte informazioni su come tecnicamente [tomb4](#) gestisce il mondo 3D e le animazioni.

In [\(Figura A\)](#) è possibile vedere l'asse 3d relativa usata in [tomb4](#).

In questo tutorial si trova anche un elenco completo dei valori [IDS](#) Stato utilizzato in [tomb4](#), e molti altri suggerimenti sulla risoluzione dei problemi con il comando di animazione e il comando [TestPosition](#).

B) la maggior parte degli esempi sono solo delle descrizioni teoriche o un comando [TestPosition](#), in ogni caso si può trovare in questa Demo anche un esempio funzionante di abilità di combattimento per [Lara](#).

In [\(Figura B\)](#) non utilizzando bene l'animazione, in questo livello, Lara può uccidere un nemico solo con un bel po' di pugni.

C) Molti esempi mostrano come utilizzare il nuovo comando [TestPosition](#), utile per rilevare quando Lara è di fronte a qualche oggetto e per interagire con esso.

In [\(Figura C\)](#) si può vedere come utilizzare i [messaggi di diagnostica di TRNG](#), per scoprire i valori critici da digitare nel comando [TestPosition](#) per scoprire quando [Lara](#) è nella posizione desiderata.

D) Molti esempi cercano di descrivere l'uso del comando di animazione, e in modo particolare, come impostare le costanti corrette ([ENV_ \(ENViroment\)](#)) per rilevare se la situazione in gioco è valida per iniziare l'animazione personalizzata.

In [\(Figura D\)](#) si può vedere quando la condizione [ENV_HOLE_IN_FRONT_CEILING_CLIMB](#) è attiva.

Dal momento che ci sono decine di [condizioni ENV](#), ho cercato di concentrare la mia attenzione solo sulle condizioni più complesse, cercando di spiegare al meglio con il comando dello Script corretto e le immagini.

FINE DEMO

Demo by Paolone traduzione in Italiano by Franky

ScreenShots



Descrizione

Barche (Boats)

[Download: Boats_new.zip \(1.78 MB\)](#)

In questa Demo troviamo un gommone e un motoscafo nello stesso livello.

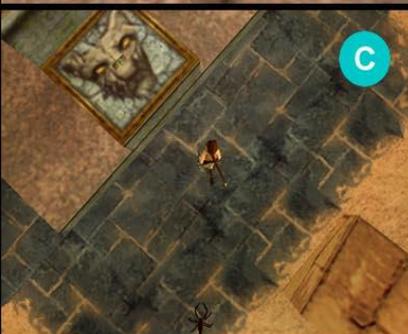
Il motoscafo ha la luce solo perché il suo **OCB=1**, ma anche con la barca si può fare la stessa cosa, se lo si desidera.

1) Questo progetto è stato modificato rispetto a quello del 2008 ora non c'è più nessun comando **AssignSlot** in **Script.txt**, e il wad dispone di quattro nuovi slot a host per il gommone e il motoscafo con i rispettivi slot per l'animazioni di lara.

2) Il Bug riscontrato sul suono, e cioè quando Lara nuota sott'acqua, è stato risolto, ma è necessario utilizzare la versione **1.2.0.1** o superiore per compilare lo **Script.txt** di questa Demo.

FINE DEMO

Demo by Paolone traduzione in Italiano by Franky

| ScreenShots | Descrizione |
|---|--|
|  | <p>Demo Telecamere (Cameras Demo)</p> <p>Download: Cameras.zip (101 Kb)</p> <p>Questo esempio mostra diversi modi, forniti da TRNG, per cambiare la modalità della Telecamera nel gioco.</p> <p>Si può guardare Lara o i Nemici da posizioni diverse.</p> <p>Si dovrebbe usare il TGA di default e gli oggetti (.Wad) del livello Karnak per questa Demo.</p> |
|  | <p>Ricordarsi di costruire lo Script.txt, copiare lo Script.txt e English.txt nella cartella Script e quindi avviare NG_Center (oppure fare clic sul pulsante [Reload] se è stato già aperto) e costruire lo Script.</p> <p>Tutti i Trigger per gli effetti della Telecamera sono posizionati sui settori con una Texture che mostra una faccia di gatto.</p> <p>NOTA: Questo esempio è stato modificato rispetto alla vecchia versione del 2008.</p> <p>All'inizio, ci sono alcuni nuovi esempi sull'utilizzo del nuovo Trigger Flieffect: "Camera. Change camera mode with <&>Parameters for (E)time"</p> |
|  | <p>Questo metodo è più facile da utilizzare, ma ha lo svantaggio di non poter essere usato per Enemies (i nemici), ma solo per Lara.</p> <p>All'inizio ci sono esempi sulla nuova funzione SET_CAMERA.</p> <p>Con questo metodo è possibile modificare la normale Telecamera, che segue sempre Lara, impostando una posizione e un angolo diverso per ognuno, in modo da guardare Lara da una posizione diversa.</p> <p>In (Figura A) è possibile vedere Lara vista dalla parte anteriore dal basso.</p> <p>In (Figura B) c'è un'altra inquadratura della SET_CAMERA, dove Lara è inquadrata dalla parte anteriore, dall'alto.</p> |
|  | <p>In (Figura C) è stato utilizzato l'altro metodo per cambiare l'inquadratura di Lara, in questo esempio viene utilizzato il metodo per imporre una vera e propria Telecamera fissa (Fixed Camera) inserita nel livello per seguire Lara e la sua definizione di una data posizione.</p> <p>In (Figura C) si può vedere come la telecamera fissa segue Lara dal piano superiore.</p> <p>D) In (Figura D) è possibile vedere l'effetto chiamato "Ritratto di Lara", che ha lo stesso layout della (Figura B), ma è stato ottenuto con la vecchia Flieffect il Ritratto di Lara ed una Telecamera esterna, mentre in (Figura B) è stata utilizzata una SET_CAMERA.</p> |
|  | <p>Quando si vuole utilizzare il "Ritratto di Lara" è meglio usare la modalità SET_CAMERA perché è più facile e perché questo metodo è in grado di evitare gli ostacoli (Walls, Ceiling, Statics).</p> <p>Potremmo dire che l'effetto "Ritratto di Lara" è ormai un metodo obsoleto per guardare Lara, dal momento che la nuova Flieffect in modalità "Change camera mode" + set_camera è più efficace.</p> <p>In (Figura E) si può vedere l'effetto dell'inquadratura di un nemico.</p> |



Questo effetto inquadra il nemico dal davanti e segue il suo movimento da questa inquadratura.

La Telecamera potrebbe avere dei **problemi** in spazi molto stretti, per questo motivo vi suggerisco di studiare molto bene dove, come e quando utilizzare questo tipo di effetto.

In (Figura F) si può vedere un "Effetto Matrix", il nome nasce dal famoso film "Matrix".

La Telecamera gira velocemente intorno mentre Lara resta ferma.

Potrebbe essere un buon sistema per mostrare finalmente il viso di **Lara** e nello stesso momento della scena di livello.

Vi suggerisco di **utilizzare l'insieme dell'effetto Matrix con la disattivazione della tastiera per forzare Lara a restare ferma**, perché se Lara si muove il risultato non è buono, ma solo un pò caotico.

In (Figura G) si può vedere l'effetto "Follow-me camera effect" per un nemico.

Questa Telecamera funziona come la Telecamera che segue **Lara**, ma in questo caso inquadrerà solo il nemico.

NOTA: tutti gli effetti hanno come bersaglio un nemico e dovrebbe essere utilizzato solo per pochi secondi, perché fino a quando saranno inquadrati, il giocatore non avrà la possibilità di comprendere la posizione di **Lara**.

FINE DEMO

ScreenShots



Descrizione

Oggetti Chronicles (Chronicles Objects)

[Download: Chronicles.zip \(3,8 MB\)](#)

Questo progetto richiede una **DLL** con la versione **1.2.1.7** o successiva.

Da questa versione sono stati aggiunti alcuni oggetti di **TR Chronicles**.

In **(Figura A)** si può vedere una corda tesa.

Impostando i valori corretti dell'**OCB** è possibile impostare il grado di difficoltà necessario per passare sulla corda: **-1**=difficoltà per i bambini (Lara non cade), **0**=difficoltà standard come in TR Chronicles, **1**=difficoltà alta, **2**=difficoltà molto alta, **3**=difficoltà impossibile.

In **(Figura B)** si può vedere il Guardiano, chiamato anche come **(Laser_Head)**.

Posizionare il Guardiano richiede molta cura, leggere il file **Readme.txt**.

Gli scatti della testa del Guardiano e i raggi laser verdi che escono dai suoi occhi uccidono Lara immediatamente.

In **(Figura C)** c'è Hydra.

Hydra lancia diverse palle di Energia (**impostabili tramite OCB**) e il suo becco uccide Lara.

In **(Figura D)** vi è il nemico sub-marine.

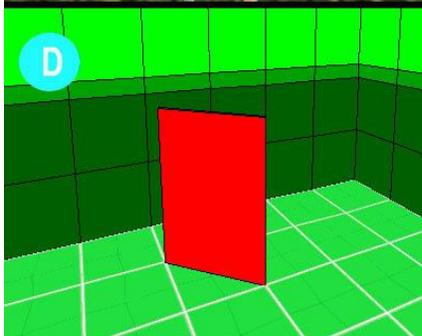
Si muove nell'acqua e cerca di colpire e uccidere Lara con un siluro.

Nel file zip troverete anche un livello giocabile con tutti questi nuovi oggetti e con tutti i nuovi suoni utilizzati.

FINE DEMO

ScreenShots

Descrizione



Collisioni sotto TRNG (Collisions under TRNG)

[Download: Collisions.zip \(3.4 Mb\)](#)

A partire dalla versione **1.2.1.5** sono disponibili le nuove funzionalità di collisione in **TRNG**.

NOTA: se siete interessati alle collisioni, guardate anche la demo **Miscellanous 2** in questa pagina, dove si possono trovare le nuove collisioni.

Ora è possibile creare collisioni invisibili nel tuo livello e, con questo trucco, è possibile consentire a **Lara** di camminare come in **(Figura A)** o salire come in **(Figura B)** su oggetti **Static** e **Moveable**.

Grazie a questa possibilità si può anche creare una cabina come quella in **(Figura C)**, dove **Lara** può entrare, trovando la corretta collisione su tutte le sue pareti, nonostante sia sospesa in aria ed è stata fatta solo con oggetti statici e collisioni invisibili.

Nel file **collis.wad** si trovano anche una serie di nuovi oggetti. **i pannelli di collisione**.

(Figura D) Si tratta di una serie di pannelli invisibili per integrare le collisioni create con i nuovi pannelli di "collisione. ..." **Flipeffects**.

I nuovi pannelli di collisione hanno un proprio nome Slot nel Wad: **PANEL_BORDER**, **PANEL_MIDDLE**, **PANEL_CORNER**, **PANEL_DIAGONAL**, **PANEL_STRIP**, **PANEL_HALF_BORDER1**, **PANEL_HALF_BORDER2**, **PANEL_MIDDLE_CORNER**.

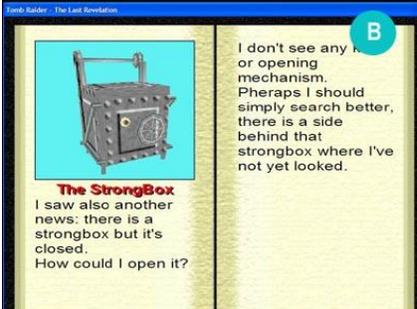
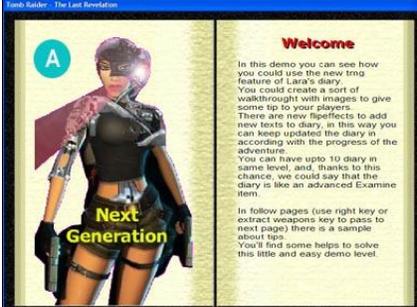
Con i pannelli, di cui sopra, è possibile creare, facilmente, le collisioni diagonali e altre forme dettagliate.

Per l'esempio della cabina **(Figura B)** ho usato i pannelli invisibili solo per arrotondare la collisione del balcone, mentre le altre collisioni sono state create utilizzando i Trigger **Flipeffect**.

In questo file zip c'è, anche, un **Tutorial Html** dove viene indicata l'esatta posizione sulla nuova funzionalità delle collisioni.

FINE DEMO

ScreenShots



Descrizione

Diario e Immagini (Diary and Images)

[Download: Diary Images.zip \(5.9 MB\)](#)

Questo esempio mostra come utilizzare l'oggetto **Diario (Diary)** vedi [\(Figura A e B\)](#) che mostrano le immagini nel gioco [\(immagine C e D\)](#).

Con il **Diario di Lara** è possibile visualizzare, testi e immagini formattati in modi diversi.

È inoltre possibile aggiungere nuove pagine usando alcuni **Trigger Flieffect**, per tenere il Diario aggiornato con i progressi dell'avventura.

Per attivare e personalizzare il Diario è necessario utilizzare il nuovo comando **Script** chiamato **Diary =**.

Si può scegliere di usare il **Font di Windows**, i **colori** e il **layout** sulla posizione del Testo nel riquadro e dell'immagine.

A proposito di immagini, ci sono due diversi tipi di immagine: le immagini pop-up [\(Figura C\)](#), in cui l'immagine copre solo una piccola parte dello schermo, mentre il gioco va avanti, e le immagini sovrapposte [\(Figura D\)](#).

Le immagini sovrapposte di solito coprono tutto lo schermo e il gioco rimane bloccato fino a quando l'immagine rimane sullo schermo.

In questo esempio è importante leggere il file **"Read_me.txt"**, perché ci sono alcune informazioni circa l'installazione delle immagini **.BMP**.

Le immagini **.BMP**, utilizzate da **TRNG**, devono essere sempre copiate nella sottocartella **PIX** di cartella **TRLE** e devono essere in **formato bitmap (BMP)**, RGB con nomi come **Image1.bmp**, **"image5.bmp"** ect.

NOTA: alcune immagini di questo esempio sono in formato **jpg** per ridurre le dimensioni del file zip, si possono convertire in formato corretto **BMP** semplicemente utilizzando la utility **Converter.exe** nella stessa cartella **PIX** dell'immagini.

Nel tuo livello sei libero di aggiungere questa utility per convertire le immagini in formato **jpg**, dando un modo semplice ai giocatori finali di riconverterli in formato **BMP**.

Se avete **TRLM 2017** si può giocare questa Demo semplicemente aggiungendo questo file zip in **TRLM** con il comando **Aggiungi nuovo livello** nel modo consueto.

Non considerare il file **"install_level.txt"** che si trova nel file Zip, è usato solo da **TRLM 2017** per installare correttamente questo livello

FINE DEMO

ScreenShots

Descrizione



Ascensori e Rivelatori (Elevators and Detectors)

Download: [Elevator_Detectors.zip \(5.98 Mb\)](#)

Questa demo contiene quattro progetti per illustrare l'utilizzo di:

A) Ascensore Stop and Go.

Questo ascensore non ha il tastierino, funziona senza fine su e giù, fermandosi solo per qualche secondo per ogni piano, non ha porta di apertura e chiusura.

B) Tastiera Ascensore.

Questo ascensore è più tradizionale, Lara entra e sceglie con la tastiera il piano la porta si chiuderà e l'ascensore si muoverà fino a raggiungere il piano desiderato.

C) Pointer Detector.

Il **Pointer Detector** è un rivelatore ed è completamente nuovo, per la saga di Tomb Raider.

Questo oggetto lavora come un Radar avanzato, permette di scoprire la posizione degli oggetti a (**nord/sud/est/ovest**), rispetto al Lara, l'altezza e la distanza in metri di qualsiasi oggetto.

Nello Script è possibile impostare la modalità di destinazione e di lavoro.

Il **Pointer Detector** permette di monitorare solo un singolo oggetto per volta, ma quando il primo obiettivo è stato raggiunto o ucciso, il rivelatore può iniziare a monitorare un altro oggetto dell'elenco digitato nel file **Script.txt**.

D) Radar Detector.

La differenza principale rispetto al **Rivelatore del Radar** e al **Pointer Detector**, è che, con il **Radar Detector** è possibile monitorare molti obiettivi contemporaneamente, quando entrano nel campo della distanza impostata nel file di **Script.txt**.

Osservazioni circa rivelatori:

1) L'oggetto rivelatore è sempre nello slot **Quest1**, di solito usato nel livello "Amuleto di Horus".

2) Se desideri spostare il **Radar Detector** dal tuo Wad ricordati di copiare anche lo slot **MISC_SPRITES**, perché contiene alcuni piccoli **Sprite** utilizzati per mostrare gli oggetti trovati sul rivelatore (quelle piccole luci lampeggianti)

Leggere per maggiori informazioni la descrizione del comando Script del Detector.

Osservazioni circa Ascensori:

1) L'ascensore "stop and go", è dotato di pannelli di collisione invisibili collocati intorno ad esso per evitare che Lara possa cadere.

Ci sono due tipi di pannelli invisibili: il **Animating1** e **Animating1_Mip**.

La differenza è la loro dimensione, il primo ha settore dimensioni 1x2 (2 è l'altezza in verticale), mentre l'altro ha una dimensione di 1x1 settori.

2) "Keypad Elevator" non ha nessuna porta in movimento, mentre l'ascensore "Stop and go" ha una falsa porta (simulata) utilizzando **Animating2**.

Per maggiori informazioni su Ascensori leggere la descrizione del comando **Script**.

FINE DEMO

Demo by Paolone traduzione in Italiano by Franky

ScreenShots



Descrizione

Pesci e Pirahnas (Fish and Pirahnas)

[Download: Fish and Pirahnas.zip \(3 MB\)](#)

L'oggetto pesce è un **emettitore**, come l'emettitore di locuste o piccoli coleotteri.

Posizionarlo nel progetto, impostare via **OCB** il numero di pesci che si desidera ottenere e quindi attivarlo in gioco per generare il numero desiderato e il tipo di pesce.

Prego, notare che questo pesce emettitore in realtà non è la stessa cosa che avete visto in **Tomb Raider 3**, ma si tratta di un nuovo oggetto creato da me (Paolone) per riprodurre i comportamenti simili a quelli visti in Tomb Raider 3.

Ci sono alcune differenze tra il pesce creato da me e il pesce di TR3, con i pesci TRNG è possibile scegliere tra quattro diversi tipi di pesce: **Pirahnas**, **Pesci Pagliaccio**, **Pesci farfalla** e **Pesci Angelo** è possibile impostare il comportamento del pesce all'interno di un Banco o come singoli individui.

È possibile scegliere anche altri comportamenti di pesci, essi saranno in grado di saltare temporaneamente fuori dall'acqua o se attaccheranno o meno Lara.

Studiare l'**OCB** per l'oggetto **FISH_EMITTER** nel pannello di riferimento del programma di **NG_Center**, nella sezione "**OCB List**" per conoscere tutti i valori **OCB** disponibili per i pesci.

In questo progetto si possono trovare tre dimostrazioni sull'utilizzo del pesce: **il primo esempio**: è per pesci **tropicali inoffensivi**, **Pesci Farfalla**, **Pesci Angelo** e **Pesci Pagliaccio** che nuoteranno vicino a Lara senza ferirla.

Il secondo esempio: mostra i pesci **Pirahnas** che naturalmente saranno mortali per **Lara**.

Il terzo esempio: mostra alcuni **pesci tropicali in un acquario**, in questo caso si vede anche il modo per costruire un piccolo acquario in una stanza.

Per i pesci d'acquario ho usato un valore di **OCB** basso (**SLOW**) per ottenere un pesce compatibile con lo spazio d'acqua molto ridotto.

Passando sulle Texture blu si possono attivare i tre diversi esempi in questo livello **demo**.

Nota: in questa demo si può trovare, anche un esempio sull'utilizzo nella grafica del **contatore delle munizioni**, sulla sostituzione dell'oggetto **Horizont** e sulle nuove **collisioni**.

FINE DEMO

Demo by Paolone traduzione in Italiano by Franky

ScreenShots



Descrizione

Filmati Video e Cutscene (FMVs: Video Cutscenes)

[Download: video_fmv_cutscenes.zip \(22 MB\)](#)

Nota: seguire le informazioni di installazione nel file [Read_me.txt](#) che avete trovato nel precedente file zip.

Questa demo ha una grande dimensione (22 Mb) perché contiene tre video **FMV** di circa 1 minuto ciascuno.

Dal momento che l'uso comune di **FMVs** in gioco è facile, ho cercato di utilizzare questo progetto di esempio per spiegare alcuni piccoli trucchi per raggiungere obiettivi comuni sulla gestione **FMV**.

1) Avviare il video prima di cominciare di livello

NOTA: vedere la [Room0](#) per questo esempio.

Se si vuole avere un video di partenza prima del livello, è possibile vedere l'esempio di prima **FMV**, questo filmato verrà riprodotto prima che il livello sia visibile.

2) Eseguire un cambiamento nel gioco quando un filmato è stato completato

NOTA: vedere [Room0](#) per questo esempio.

NOTA: Nella maggior parte dei casi si usa un video per illustrare alcuni fatti importanti accaduti nel gioco, per questo motivo alla fine del **FMV** dovremmo impostare qualche cambiamento in gioco per dare un senso al nostro video.

3) Il video alla fine del livello attuale e prima del caricamento di un altro livello.

Quando si intende avere un **FMV** finale per il vostro livello e avere il caricamento del livello successivo immediatamente alla fine della **FMV**, si può seguire l'esempio nella [stanza 54](#).

FINE DEMO

ScreenShots

Descrizione



Nebbia Distanza Esercitazione (Fog Distance Tutorial)

[Download: Fog_Distance_Tutorial.zip \(381 Kb\)](#)

A partire dalla versione **1.2.1.0** alcuni Bugs sulla nebbia a distanza sono stati corretti.

In questo **Tutorial** si possono trovare **multi esempi e suggerimenti** sulle impostazioni corrette per la nebbia a distanza.

In (**Figura A e B**) nel file **Html** ci sono le immagini con impostazioni diverse.

C'è una introduzione anche di **nuovi Flipeffect** per creare effetti di nebbia dinamica per simulare una tempesta nel deserto (Desert Storm).

In (**Figura C**) troverete, anche, tutte le informazioni su come correggere il **Bug** sulla saturazione delle Texture trasparenti con il colore della nebbia.

In (**Figura D**) utilizzando il nuovo fissaggio è possibile rimuovere quel Bug e quindi è possibile utilizzare liberamente la nebbia a distanza anche in ambienti con acqua.

FINE DEMO

ScreenShots



Descrizione

Arpione e Sub Man (Harpoon e Frog-Man)

[Download: Harpoon.zip \(2.4 Mb\)](#)

A partire dalla versione **1.2.1.8** è disponibile per Lara l'**Arpione** di Tomb Raider 3 (**Figura A**).

È possibile impostare il **numero di munizioni** e il **tipo di Arpione** con il comando Script **Customize = CUST_HARPOON**, si veda anche lo **Script.txt** nella corrente Demo.

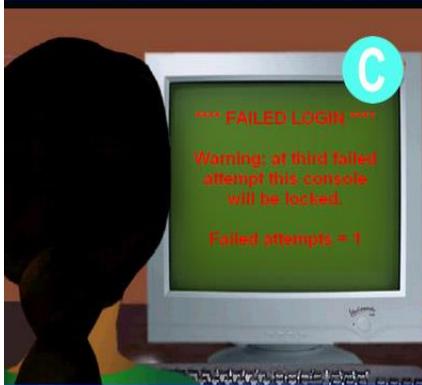
(**Figura B**) in questa demo c'è anche un'altra notizia relativa alla **DII** della versione **1.2.1.8**: il nemico **Uomo-Rana**.

Dalla versione **1.2.1.8** c'è anche la possibile di **abilitare o disabilitare il fascio di luce** della della Jeep (**Figura C**) o del sidecar, in entrambi i casi è possibile impostare queste modifiche digitando **OCB=1** nella jeep o il sidecar, mentre lasciando **OCB=0** questi veicoli funzioneranno in modo comune.

FINE DEMO

ScreenShots

Descrizione



Casella di Input e StaticMip (Input Box and StaticMip)

[Download: InputBox_StaticMip \(4.4 Mb\)](#)

In questo livello con la versione **1.3.0.0** ci sono Demo con due caratteristiche principali: **1)** la nuova casella di input, in base **PARAM_INPUT_BOX** nuovo parametro costante, **2)** il nuovo comando Script **StaticMip=**

InputBox

Con questo campo è possibile richiedere un testo dal giocatore e lui sarà in grado di digitare, **lettera per lettera o numero per numero**, il testo sullo schermo con un cursore lampeggiante per simulare una **Console di un PC**.

Pertanto, la casella di Input è come nei migliori KeyPad, ha alcune importanti differenze: il giocatore può inserire **non solo numeri ma anche lettere**.

InputBox non si basa su qualche interruttore 3D, ma funziona in grafica 2D, quindi è possibile passare questa funzione in qualsiasi tipo di immagine.

Questa demo (**Figura A**) simula una console PC dove Lara deve digitare il Login (**Figura B**).

Per dimostrare come è possibile simulare un lavoro nel gioco, impostare anche un codice tramite **Triggergroups** e **Variabili**, per simulare questo comportamento:

Il PC richiede nome utente (**Username**)

Il PC richiede la Password (**PSW**) (per quel nome utente)

Se i dati del login sono errati, il PC ricorda il numero di tentativi falliti effettuati e, dopo il terzo tentativo fallito (**Figura C**), la console verrà bloccata.

Se Lara digita i dati del login a destra, la console si ricorderà che è già registrato e, in seguito l'accesso, mostrerà immediatamente il menù dei comandi, dove Lara può scegliere un comando per eseguire qualche cambiamento nel livello.

StaticMip

Il comando Script **StaticMip =** permette di impostare un'intervallo di distanza, tra il punto di vista di origine (di solito Lara) e l'oggetto Static, dove l'oggetto statico verrà disegnato in bassa qualità, risparmiando tempo.

(**Figura D**): quando l'oggetto è molto molto lontano, tutti gli ulteriori dettagli non saranno non più ben visibili, ed è uno spreco di grafica.

Così, l'idea è buona e **StaticMip** funziona bene.

È possibile, non solo impostare un intervallo per mostrare una brutta copia dell'oggetto principale, ma anche impostare un secondo intervallo, dove, se lo si desidera, l'oggetto verrà ignorato e non disegnato.

Dall'altra parte, la cattiva notizia è che, questa Demo Static, **è un fallimento epico formidabile!**

Quando **Lara** è in grado di vedere il momento di scambio tra l'alta qualità e lo Static di bassa qualità (e in questo Demo è possibile) si vede lo Static che sembra cambiare la sua forma (confine) e, in alcuni circostanze, si muove un pò.

Il problema è che, per gestire il modo migliore per questo scambio, (momento critico), è necessario un grande lavoro e non ho avuto tutto questo tempo, così ho rilasciato la Demo Static in questo brutto modo, mi dispiace, comunque, avendo più tempo da dedicare, posso migliore questo scambio.

Ci sono due soluzioni:

1) Creazione di una bassa risoluzione (brutta copia) di uno Static principale che ha avuto, almeno, esattamente lo stesso confine (contorno), perché è la forma che, anche da molto lontano, è sempre visibile.

2) Impostare il livello, per evitare che Lara (e il giocatore) non siano in grado di vedere questo scambio. Questo è possibile posizionando in modo giusto qualche ostacolo alla vista, posizionato con giusto numero di settori, in accordo con il Range impostato nel comando [StaticMip](#).

Su mio suggerimento utilizzare oggetti [Crossed Face](#) (due rettangoli incrociati) (vedere il file [Readme.txt](#) all'interno del file zip), i risultati in questa Demo sono davvero deludenti, comunque... sono sicuro che, con altri oggetti (meno irregolari come le piante) e con alcuni avendo più cura, potrebbe essere che, anche tale suggerimento, una buona idea.

FINE DEMO

Demo by Paolone traduzione in Italiano by Franky

ScreenShots



Descrizione

Jeep e Sidecar (Jeep and Side-Car)

[Download: JeepBike.zip \(2.1 MB\)](#)

La **Jeep** e il **Sidecar** non sono una grande novità nel editor di livelli di Tomb Raider, in questo caso la notizia è che l'utente può utilizzare **entrambi i veicoli nello stesso livello**.

Per ottenere questo risultato è necessario inserire un comando **Script** per spostare chiave della Jeep in altri Slot, per evitare la sovrapposizione con la voce di Protossido di Azoto, usato dal Sidecar.

Mentre per altro conflitto: il veicolo slot aggiuntivo utilizzato per memorizzare l'animazione di Lara per entrambe i veicoli, ora è **possibile inserire le animazioni Lara, per il sidecar**, in un nuovo slot.

NOTA: In questo modo è possibile avere entrambi i veicoli nello stesso livello.

FINE DEMO

Demo by Paolone traduzione in Italiano by Franky

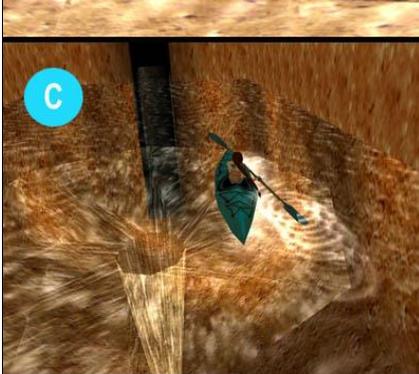
ScreenShots



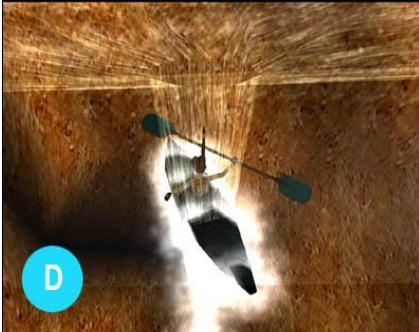
A



B



C



D

Descrizione

Canoa (Kayak)

[Download: Kayak.zip \(3.6 MB\)](#)

Il **kayak** è stato importato da **Tomb Raider III**, dal livello **Mudubu**.

In **(Figura A)**: con il **kayak**, diversamente dal motoscafo e dal gommone, è possibile discendere le **rapide di un fiume**.

In questo livello si trovano **due diversi tipi di rapide**, con **cascata trasparente (Figura A)**, o **cascata opaca (Figura B)**.

È possibile impostare un **OCB** diverso per il **kayak** per abilitare alcune caratteristiche, come il **"look-at"** cioè la caratteristica di vedere **l'effetto nebbia** sulle rapide o in movimento normale.

In questa **Demo** si trova anche un esempio del nuovo **effetto vortice (Figura C)**, dove Lara (con kayak, Barca a motore, Gommone) potrebbe essere risucchiata dal vortice uccidendola **(Figura D)**.

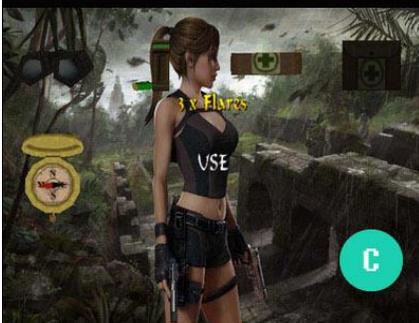
Alcuni codici **OCB** del **kayak** sono utilizzati per consentire nuove **animazioni personalizzate per salire a bordo**.

Il discorso è un pò complesso, quindi se siete interessati, vi consiglio di leggere attentamente gli **OCB** per il kayak descrizione pannello di riferimento del programma di **NG_Center**.

NOTA: Come per le altre demo, è possibile utilizzare **tutti gli oggetti** che si trovano nei Wad.

FINE DEMO

ScreenShots



Descrizione

Lampi e Cutscene (Lightnings and Cutscenes)

Download: lampi e binocolo (Fonti) (16 MB) (NON ESISTE)

Download: Cutscenes and Demos - Binary, playable demo (53 MB) (NON ESISTE)

Il rilascio della versione 1.2.2.7 ha avuto un tempo di produzione lungo e per questo motivo vi è una storia molto lunga per il **fissaggio di molti Bug**, e l'aggiunta di nuove funzionalità.

In questa pagina c'è solo un piccolo riassunto di queste notizie.

Dalla versione 1.2.2.7 di TRNG, è stato aggiunto **l'effetto fulmine**.

È possibile personalizzarlo in molti modi:

- 1) per farlo funzionare come i **lampi** dal cielo in **modo casuale** (Figura A)
- 2) come **conduttore elettrico** tra due punti (Figura B).

Con la versione 1.2.2.7 è anche possibile aggiungere **immagini di sfondo alla schermata** del livello di carico, schermata dell'inventario e mettere in pausa / opzioni/schermata Statica (Figura C).

C'è una **nuova personalizzazione** per il binocolo (Figura D), dove è possibile, aggiungere una nuova **maschera di grafica**, visualizzando informazioni su dati della bussola, Sestante, per scoprire la rotazione verticale della vista, fattore di zoom e sensori di luminosità.

Inoltre è possibile impostare **alcuni flag**, per personalizzare il **binocolo**, per modificare la velocità dello zoom **In** e **Out** e per aumentare il **fattore di zoom massimo**, il vecchio era circa 4x, il nuovo di 40x.

Ci sono molte altre nuove Funzionalità con la versione 1.2.2.7 le più importanti sono le seguenti:

*** Aggiunta nuova finestra di setup in tomb4.exe:**

ora è possibile avviare il programma di installazione, tenendo premuto il tasto **CTRL** mentre il gioco è in fase di avvio.

Nella finestra di configurazione ci sono **nuove impostazioni di emergenza** per risolvere i problemi con immagini, filmati, rallentamento o tremolanti.

Alcune di queste impostazioni di emergenza sono molto importanti per risolvere i problemi con **8 sistemi operativi Windows7** e immagini o FMVs.

*** Aggiunto qualche direttiva per NG_Center:**

per includere file (**#INCLUDE**), definire costanti temporanee (**#DEFINE**), per ottenere più pulito e ben ordinato lo **Script**.

*** In Ngle (new room editor) (nuovo editor di Stanze) è stata aggiunta una nuova finestra personalizzabile:**

dove è possibile aggiungere i pulsanti per riprodurre le funzionalità sui comandi noti del programma, usando i tasti come le **scorciatoie** per la tastiera o comandi di menù per farli più facile e veloce.

*** In NG_Center è stato incrementato il numero massimo di occorrenze**

per molti comandi Script: **TriggerGroup, GlobalTrigger, Organizer** ect.

In pratica ora è possibile digitare un maggior numero di questi comandi nella stessa sezione di livello.

*** Ora è possibile utilizzare charset orientale a livello di Tomb Raider:**

Questo significa che è possibile utilizzare i caratteri Cinesi, Giapponesi, Turco o Russo (e molti altri) per mostrare i testi in gioco.

*** Aggiunti nuovi effetti di testo:**

dove il testo inizia con carattere molto piccolo e le sue dimensioni crescono velocemente, modifica del formato grande/piccolo in modo ritmico lampeggiante, o cambiando colore (il colore del testo verrà modificato con diverse sfumature di colori in modo più veloce e facile).

*** Fissati molti Bug su:**

Ascensore, [CUST_KEEP_DEAD_ENEMIES](#), esplosione di cani e mummie, estensione del numero di luci dinamiche (ora funziona con **127 luci** contro i vecchi 32 luci), piccolo pulsante (**inutile**) sopra la finestra [NG_Center](#), binocoli con comandi da tastiera per l'inventario, schermata di pausa, dimensioni del carattere del [Diario](#), che mostrano le immagini, la modalità LaserSight, flyby trigger

*** E' stato Aggiunto un set completo**

Esso è stato aggiunto un set completo di **nuove funzionalità** per gestire le scene ([Figura E](#)): ora è possibile cambiare i capi di Lara e altri nemici per simulare il parlare o di altri stati d'animo. È possibile utilizzare una Telecamera "cutscene" per cambiare rapidamente e visualizzare facilmente i diversi attori.

La nuova demo permette di mostrare in anticipo, dal titolo molte suggestive scene della vostra avventura.

*** E 'stato aggiunto al pannello "Tools3"**

lo **strumento di conversione audio** molto utile per convertire i file wav in formato giusto per la cartella [sound/samples folder](#), per convertire i sound da **MP3** e **WAV** e viceversa.

*** In Ngle è stata aggiunta una nuova impostazione**

per migliorare la velocità sulla costruzione delle stanze in modalità [Draw Doors](#).

FINE DEMO

ScreenShots



Descrizione

Specchi (Mirrors)

[Download: Mirrors.zip \(1.04 Mb\)](#)

[NOTA: puoi anche consultare i: nuovi specchi \(3.19 MB\) demo.](#)

In questa Demo, è possibile vedere alcuni tipi di Specchi (Mirrors) che è possibile creare con TRNG.

(Figura A) Specchi Orizzontali.

Questo specchio è molto simile al vecchio specchio, che si può vedere nel livello **Costiera**, ma in questo caso è possibile duplicare anche altri oggetti, e saranno spostati e animati in modo speculare, come se fossero **davvero riflessi dallo specchio**.

In (Figura A) Lara sta spingendo un **oggetto pushable** mentre nella sala degli specchi un'altra Lara spinge lo stesso oggetto pushable.

Nella stessa (Figura A) si può vedere la mummia, nella sala principale e nella **camera riflessa**.

Per realizzare questo obiettivo è necessario un corretto set di comandi **Mirror** nello Script si digitano **gli indici degli oggetti** della stanza principale, per duplicarli nella stanza nascosta.

Per maggiori informazioni consultare la descrizione del comando **Script** per lo specchio (**Mirror**).

In (Figura B) Specchio Inverso Orizzontale.

Nella (Figura B) è possibile vedere un altro specchio orizzontale, ma in questo caso la **riflessione è invertita**, come visto nel vecchio **Tomb Raider 1**, quando Lara incontra il suo clone alieno.

Anche questo tipo di specchio è possibile crearlo impostando gli oggetti come **(lo scheletro in Figura B)**, anche loro saranno riflessi in modalità inversa.

C) Specchio sul Pavimento.

Nella (Figura C) si può vedere uno specchio sul pavimento.

D) Specchio sul Soffitto.

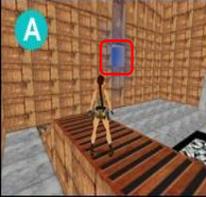
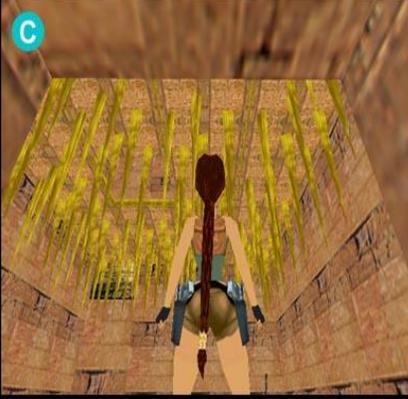
Nella (Figura D) è possibile vedere lo specchio sul soffitto.

Questo specchio è un **po' noioso**, perché con il comando **sguardo** di Lara **non è in grado di vedersi riflessa sul soffitto**.

Se si desidera utilizzare questo tipo di specchio, **suggerisco di modificare** il piano ottico in quella situazione, utilizzando una telecamera fissa (**Fixed Camera**) o di qualche nuova funzionalità **TRNG** per inquadrare **Lara** da una posizione diversa, per arrivare a vedere in modo migliore la vera Lara riflessa contemporaneamente.

FINE DEMO

Demo by Paolone traduzione in Italiano by Franky

| ScreenShots | Descrizione |
|---|--|
|   | <p style="text-align: center;">Miscellaneous</p> |
|  | <p>Scarica: Mis Sample 1.zip (2,9 MB)</p> <p>Si tratta di una Demo giocabile, dove si devono superare alcune trappole e difficoltà.</p> <p>In questa Demo ci sono le dimostrazioni sull'utilizzo dei seguenti argomenti:</p> <p>A) Come utilizzare UV scorrimento delle Texture e nuovi Trigger Pysics, per simulare un tapis roulant, in (Figura A), Lara è spostata dal tapis roulant.</p> <p>Come creare un metodo di salvataggio come nel tomb raider 3, dove per salvare il gioco era necessario trovare un cristallo blu da utilizzare quando si desidera salvare il gioco.</p> <p>Se si guarda attentamente, nella (Figura A), si può vedere un foro nel muro con luci blu, bene in quel buco c'è un cristallo blu, l'unico modo per salvare il gioco in questa Demo.</p> <p>Studiare lo Script.txt per scoprire come questo metodo è stato realizzato. È necessario rimuovere l'oggetto comune, salvare la scheda di memoria dall'inventario (questo disabilita anche il tasto F5 per salvare il gioco) e quindi creare un GlobalTrigger per eseguire un salvataggio solo quando Lara seleziona dall'inventario un determinato oggetto: il cristallo blu.</p> |
|   | <p>B) come utilizzare Trigger Frammentati per creare una griglia sensoriale. Nella (Figura B) si può vedere il risultato finale, Lara deve evitare di passare sulle linee giallo/rosso perchè sarà infortunata.</p> |
|   | <p>È possibile modificare questo esempio anche per consentire ad altri fattori scatenanti, ad esempio invece di bruciare Lara, di coinvolgere un sistema di allarme, o sostituire il semplice tratto di strisce con Static 3D come raggi laser rossi.</p> <p>In (Figura C) si vede come utilizzare il Trigger per spostare gli elementi (oggetti Moveable) e con il comando Script ItemGroup, per creare un soffitto in movimento con lancia (Figura C) si cede il soffitto che si muove senza fine SU e Giù.</p> |
|   | <p>D) Come modificare il "Tempo" in fase di esecuzione.</p> <p>In (Figura D) e in seguito, fino all'immagine K, si può vedere il cambio eseguito nel gioco utilizzando un "Organizer Script command", comando per sincronizzare i tempi per coinvolgere diversi Trigger per modificare il tempo.</p> <p>Il cielo comincia da quasi chiaro (Figura D) a diventare torbido, inizia una leggera pioggia (non molto visibile in (Figura E e F)) e immagini di tuoni e bagliori, alla fine inizia a nevicare.</p> |
|   | <p>Un'altra dimostrazione interessante, in questo esempio, è su come creare l'effetto della neve che, progressivamente, fino ad ottenere un prato bianco che, all'inizio, era verde (Figura D), e poi diventa completamente bianco (Figura K)</p> <p>Questo effetto è stato ottenuto utilizzando P_Frame, dove la singola Texture è applicata al piano, sono state modificate dopo un certo intervallo di tempo, con altre Texture animate più bianche.</p> |
| | <p>FINE DEMO</p> |

ScreenShots

Descrizione



Miscellaneous II

[Download: Miscellaneous II \(4 MB\)](#)

Prima di giocare questa Demo si deve ricostruire file **.Dat** e **.Tr4**.

In questa demo si trova una dimostrazione delle nuove collisioni ponte con un [Tutorial di Html](#) che mostra le caratteristiche, trucchi e consigli per ottenere migliori risultati con questo nuovo tipo di collisione.

Grazie al ponte miscelato con [Condizional Fragment Trigger](#), è possibile creare oggetti appesi su passerelle con nuove forme.

In [\(Figura A\)](#) è possibile vedere un anello su cui **Lara è in grado di camminare** senza cadere nel foro centrale.

Aumentando la profondità (altezza) della collisione ponte, (e si può farlo utilizzando i nuovi **OCBs** per ponti), è possibile utilizzare un ponte invisibile per simulare una collisione precisa di qualche oggetto, come la colonna [\(Figura B\)](#).

Tale colonna ha una collisione rotonda dove Lara è in grado di camminarci sopra o intorno.

In [\(Figura C\)](#) Un'altra funzionalità del ponte è quella di creare una pendenza con inclinazione maggiore di 2 click e dove Lara è in grado di camminarci senza

In effetti, ora, ci sono nuovi oggetti: la [BRIDGE_TILT3](#), [BRIDGE_TILT4](#) e la [BRIDGE_CUSTOM](#) dove è possibile impostare il grado di inclinazione utilizzando alcuni valori **OCB**.

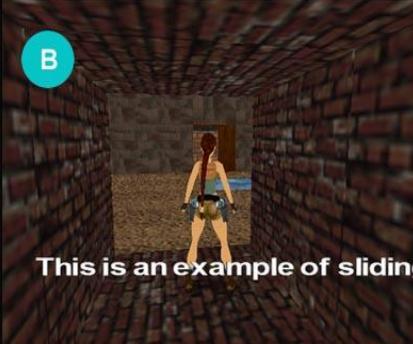
Altri le collisioni ponte, in questa **Demo** si trovano anche alcuni esempi circa il (ridimensionamento) caratteristica di una nuova scala. E' ora possibile cambiare la dimensione degli oggetti statici, utilizzando i valori **OCB** o in modo dinamico con alcuni nuovi Trigger Flieffect. Nella [\(Figura D\)](#) il leone cambia la sua dimensione in modo dinamico. E' possibile ridurre o ingrandire qualsiasi statico in tempi diversi per velocità diverse.

In [\(Figura E\)](#) un'altra nuova caratteristica che ho mostrato nel [Miscellaneous II](#), è che grazie agli **Sprite**, ora è possibile mostrare lo **Sprite 2D sullo schermo** come se fossero piccole immagini.

È stato aggiunto un nuovo oggetto **Sprite** denominato **CUSTOM_SPRITES** dove è possibile memorizzare molte immagini per mostrare sullo schermo in modo singolo come ottenere una griglia con immagini più ampie.

Gli **Sprite**, rispetto alle **Flieffects**, hanno il vantaggio che è possibile visualizzare più immagini in una sola volta ed è possibile utilizzare anche gli **Sprite** semitrasparenti e si possono colorare.

FINE DEMO

| ScreenShots | Descrizione |
|--|---|
|  <p>At left you can see some available size for fonts.</p> <p>Remark: the micro size and atomic char are fixed size and keep same size with different font_graphics object, while others size will be affected by original size and shape of font graphics</p> | <h3 data-bbox="776 264 1227 296">Nuovo Tipo di Carattere (New Font)</h3> |
|  <p>This is an example of sliding</p> | <p data-bbox="634 327 922 352">Download: NewFont.zip (2.5 Mb)</p> |
|  <p>My Marvellous Level</p> <p>This example shows how to create a multi-size or multi-color scrolling text. In the reality there are three different texts showed in sequence using the new global trigger:</p> <pre>GT_VSCROLL_LAST_VISIBLE</pre> <p>This global trigger will be enabled when a scrolling text show the last row on screen, using this signal to start new scrolling text trigger, we get the result to have two or more scrolling text showed in same time, each of them with its setting about size and color.</p> <p>Read the script.txt file supplied with this project to understand how to set the globaltrigger commands.</p> <p>Created by Paolone</p> | <p data-bbox="634 384 1367 457">Questa Demo è stata modificata, rispetto alla versione del 2008, ora è stato aggiunto anche un esempio di testo scorrevole, e il movimento di un testo a scorrimento verticale è stato migliorato.</p> |
| | <p data-bbox="634 489 1318 541">Questo esempio è importante per il file allegato Wad dove è possibile trovare il nuovo oggetto FONT_GRAPHICS.</p> |
| | <p data-bbox="634 564 1341 663">È possibile modificare il tipo di carattere utilizzando lo strumento NG Font Editor che trovate in pannello Strumenti di NG_Center, in ogni caso è necessario copiare l'oggetto FONT_GRAPHICS originale nella Wad prima di modificarlo con NG Font Editor.</p> |
| | <p data-bbox="634 688 1325 741">Nella (Figura A), in questo breve esempio, si può vedere la nuova formattazione differente disponibile per i caratteri.</p> |
| | <p data-bbox="634 772 1325 825">È possibile modificare la dimensione circa la larghezza o l'altezza, o di utilizzare alcuni tipi di carattere di dimensioni fisse.</p> |
| | <p data-bbox="634 852 1341 926">Questa Demo mostra anche due interessanti Flipeffect che si attivano per visualizzare lo scorrimento del testo in verticale (Figura C) o in orizzontale, come una singola striscia di informazioni veloci (Figura B).</p> |
| | <p data-bbox="634 957 1367 1010">Nel precedente esempio (Figura A e B) il testo è in movimento, verso l'alto (Figura C) e da destra verso sinistra in (Figura B).</p> |
| | <p data-bbox="634 1037 1357 1110">NOTA: 1) C'è un Bug noioso quando si utilizza il nuovo NG Font in un file wad c'è anche un oggetto di cascata. La Texture animata della cascata potrebbe essere compromessa e potrebbe apparire come una Texture con valori letterali (A C F ecc).</p> |
| | <p data-bbox="634 1148 1357 1201">Ho cercato di risolvere questo Bug, ma questo accade in Wadmerger, quando si crea il nuovo file .SWD con Texture.</p> |
| | <p data-bbox="634 1224 1367 1318">2) Quando si genera lo Script.dat, ricordarsi di copiare nella cartella Script anche 3 files: "Text_DisableInput.txt", "Text_FontDescription.txt" e "Tex_MultiSize.txt" perché sono utilizzati come stringa esterna e saranno messi in English.dat quando si genera lo Script.</p> |

FINE DEMO

ScreenShots



Descrizione

Nuovi Specchi (New Mirrors)

[Download: nuovi specchi \(3.19 MB\)](#)

A partire dalla versione **1.2.2.3** le caratteristiche degli specchi sono stati migliorati.

In **(Figura A)** ora ci sono nuovi specchi diversi da ovest, Pavimento (Floor) e soffitto (Ceiling), ed è possibile inserire uno specchio in una parete della stanza.

In **(Figura B)** ora è possibile inserire uno, due o più specchi nella stessa camera, studiate questa Demo per scoprire come mettere due specchi nella stessa stanza evitando il problema del multiriflesso.

Nell'esempio viene illustrato come utilizzare un oggetto Switch per eseguire alcuni Trigger speciali, come spostare un oggetto **Static** utilizzando lo stesso switch con l'attivazione del Trigger e Antrigger.

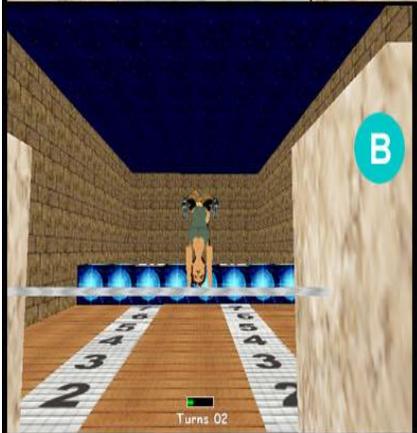
FINE DEMO

ScreenShots

Descrizione



A



B



C



D



E

Barre Parallele (Parallel-Bars)

[Download: Parallel_Bars.zip \(2.2 Mb\)](#)

Il Parallel-bars è un oggetto che abbiamo già visto in [Tomb Raider Chronicles](#).

In questo esempio vi sono due livelli, uno con la stessa [Parallel_bars](#) di TRC senza modifiche, mentre nella seconda c'è un Parallel-bars personalizzata con alcune novità.

A) In [\(Figura A\)](#) è possibile vedere la Parallel-bars di default.

La lunghezza del salto è influenzato dal valore impostato nel campo [OCB](#) della Parallel-bars nel programma [NGLE](#).

In mancanza dell'[OCB](#) nel Parallel-bars ha un'unica direzione "buona", dove il valore di [OCB](#) darà un salto in lungo, mentre in direzione opposta Lara eseguirà il salto ma con una perdita nel salto, indifferente dal valore di [OCB](#).

NOTA: se lo si desidera, è possibile rimuovere questa modalità di default di esecuzione, costringendo la Parallel_bar a funzionare nella stessa modalità per entrambi i lati.

È possibile realizzare questo con il comando:

[Customize = CUST_PARALLEL_BARS, PB_DOUBLE_FACE, IGNORE, IGNORE, IGNORE](#)

Il Flag [PB_DOUBLE_FACE](#) serve per l'esecuzione della Parallel_bar nello stesso modo per entrambi i lati.

B) La barra personalizzata è in grado di supportare diverse caratteristiche in accordo con i valori impostati nel comando Script : [Customize = CUST_PARALLEL_BARS](#).

In questa demo ho riassunto tutte le caratteristiche su misura per la Parallel_bar nel secondo livello, ma è possibile scegliere naturalmente solo alcune singole caratteristiche.

Nella [\(Figura B\)](#) è possibile vedere il risultato utilizzando il Flag:

[PB_SHOW_CHARGE_BAR](#)

e

[PB_SHOW_CHARGE_COUNTER](#)

Il Flag [PB_SHOW_CHARGE_BAR](#) mostra la barra di avanzamento con il potere di salto di corrente (la barra è di colore verde), mentre il Flag [PB_SHOW_CHARGE_COUNTER](#) mostra lo stesso potere del salto in forma di numero di giri.

NOTA: se si vuole avere un potere progressivo nel salto, dovrete anche personalizzare il comando il flag [PB_PROGRESSIVE_CHARGE](#), per attivare il metodo di ricarica, turno dopo turno.

C) Un'altra personalizzazione utilizzata è quella attivata con il Flag [PB_MULTIPLE_ENDINGS](#).

La realizzazione tecnica non è perfetta, in ogni caso l'idea è quella di dare al giocatore la possibilità di uscire dalla rotazione in posizione diversa, con risultati diversi.

Quando il giocatore lascia il tasto [AZIONE \(Action\)](#) mentre Lara è nella posizione mostrata in [\(Figura C\)](#), Lara cadrà come si vede in [\(Figura D\)](#).

E) Mentre se il giocatore lascia [AZIONE ACTION](#) in posizione mostrato nella [\(Figura E\)](#), Lara cadrà come mostrato in [\(Figura F\)](#).



In questa situazione [Lara](#) spesso [muore](#).

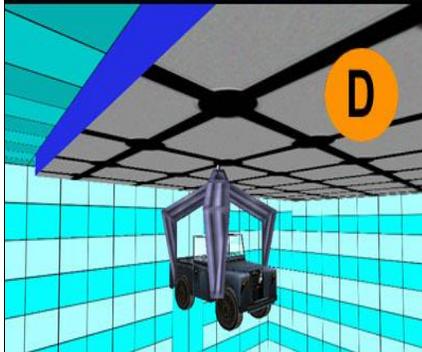
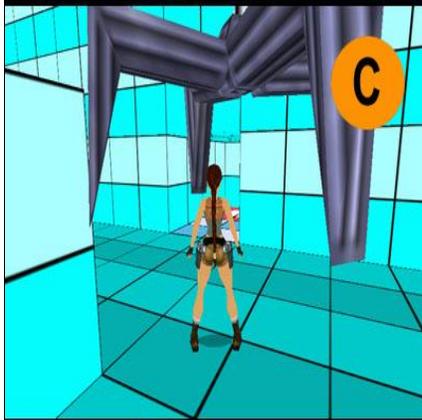
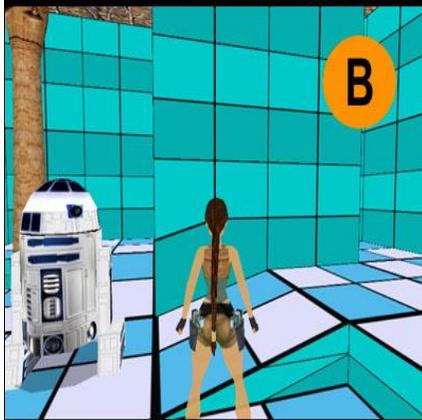
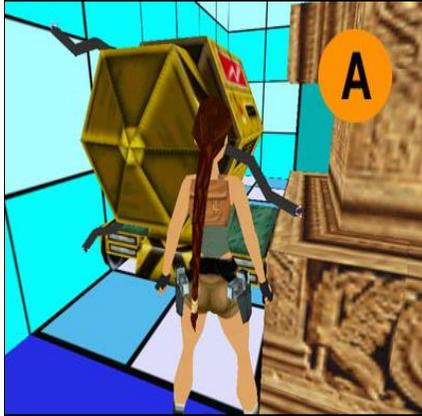
Il finale multiplo è un modo per aggiungere nuova interattività, ma le animazioni utilizzate non sono perfettamente collegate con le precedenti animazioni svolte da quando ho usato alcune vecchie animazioni standard.

Se si desidera ricostruire questi finali multipli con le animazioni personalizzate, si può fare, basta omettere il flag [PB_MULTIPLE_ENDINGS](#) per evitare conflitti, e quindi seguire le istruzioni che si trovano nella descrizione del [CUST_PARALLEL_BARS](#) costanti nel pannello di riferimento di [NG_Center](#).

FINE DEMO

ScreenShots

Descrizione



Demo Plugin (PluginDemo)

[Download: Plugin_Obj_Demo.zip \(3,9 MB\)](#)

Questo livello richiede la versione **1.3.0.0** (o superiore) e la versione **NG_CENTER 1.5** (o superiore).

Poiché si tratta del primo livello che richiede il **Plugin**, userò anche questa pagina per descrivere meglio questa nuova possibilità di **TRNG**.

In questo livello **Demo** si trovano quattro nuovi oggetti: **(Figura A)**=Cleaner Robot, **(Figura B)**=Star Wars Robot, **(Figura C)**=Gru (come trappola, **(Figura D)**=Gru guidabile e infine **(Figura E)**=(*) veicoli Mech Warrior.

(*) NOTA: Mech Warrior è stato costruito da Psiko, Mesh e Animazioni, mentre io (Paolone) aggiunto il codice per animarlo.

Il codice necessario per gestire questi nuovi oggetti non è presente in **tomb_nextgeneration.dll** ma nel loro specifico Plugin.

Per esempio per vedere in **movimento del robot pulitore** è necessario **installare** nel NG_Center il **Plugin Cleaner**, o di avere le Gru (entrambi: trappola o guidabile) è necessario **installare** il **Plugin Gru**.

Per installare in **NG_Center** un Plugin avrete bisogno dell'installazione del file Plugin, in qualche parte della **documentazione di SDK**, chiamato anche: **(Plugin runtime files)**.

Si tratta di un insieme di file che hanno tutti lo stesso nome **(Pugin_SomeName)** ma diverse estensioni.

In pratica si possono ricordare i file WAD del livello Karnak, avendo lo stesso nome ma estensioni diverse (**karnak.was**, **karnak.sfx**, **karnak.wad** ect.).

Molto simili rispetto ai file **WAD**, i file dei Plugin, hanno le estensioni diverse e saranno:

.DLL (sempre presente, è il nucleo del codice del Plugin, come **tomb_nextgeneration.dll** file è il cuore del motore di **TRNG**).

.BTN (è opzionale ma è molto importante per personalizzare il proprio pannello di Plugin programma **NG_Center** "BTN" significa file di pulsante).

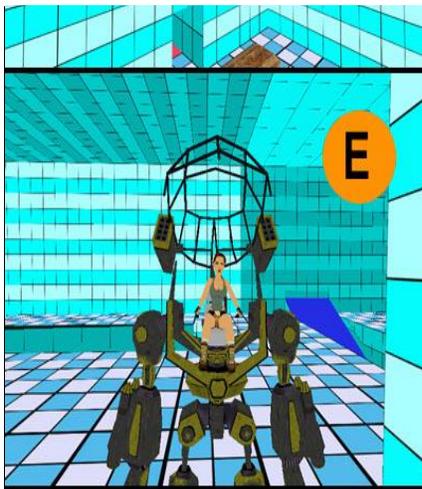
.SCRIPT (facoltativo ma necessario quando il Plugin richiede comandi o costanti alcuni una propria scrittura).

.EXE (la sua presenza non è molto probabile, comunque il file di plugin **EXE** **potrebbe essere in grado** di interagire con il programma **NG_Center** in modo diretto, per fornire funzioni di Script/strumento specifico per obiettivi)

.TRG (il Trigger file Plugin è richiesto per aggiungere nuovi trigger per il programma **NGLE**).

.OCB (quando il plugin aggiunge nuovi valori di **OCB** per i suoi oggetti, o da altri Plugin in **.OCB**, nel file ci sarà la documentazione su questi nuovi valori di **OCB**. Tutta questa documentazione è **visibile** nel pannello dei riferimenti di **NG_Center** insieme ad altre descrizioni di **OCB**).

.TXT (un file di testo con una breve descrizione del Plugin corrente, questo testo sarà caricato nel suo pannello di Plugin programma **NG_Center**).



NOTA: ricorda che, escludendo i files Exe e Dll, tutti gli altri files di configurazione di default sono file di testo comune.

Per questo motivo, nonostante le loro estensioni, si dovrebbe aprire utilizzando il [Block Notes](#) o [altri editor di testo](#) se si desidera visualizzare o modificare il loro contenuto (comunque questa operazione è consigliato solo per gli autori del Plugin, naturalmente).

In questo elenco abbiamo visto i files di installazione di default del Plugin, in ogni caso l'autore del Plugin potrebbe aggiungere altri files nel pacchetto per questo Plugin, ovviamente, e, in tal caso, non è necessario seguire la regola del formato "[Plugin_SomeName](#)".

Per esempio, può essere (ed è anche suggerito), aggiungere una descrizione estesa (come un file di aiuto) per descrivere l'uso di Plugin con una migliore e più dettagliata descrizione di quella presente nel file di [Plugin_SomeName.txt](#).

E' necessario [installare e avviare questa Demo Plugin](#), avremo per tutti i loro files un piccolo aiuto, per spiegare al costruttore di livelli come utilizzare i nuovi oggetti (o altre abilità) fornito con quel plugin.



Se si guarda pannello del Plugin Mech Warrior, per esempio, si trova un [\[Input Commands Help\]](#) [Input Comandi Guida] per avere la documentazione sui comandi della tastiera per guidare il Mech Warrior.

Mentre negli altri Plugin è presente un pulsante [\[Export Description\]](#) [Esporta Descrizione] per avere, un programma predefinito di editor di testo autonomo, la descrizione predefinita di quel Plugin vedi [\(Figura F\)](#).

NOTA: i nomi di pulsante non seguono uno [Standard o regole](#): è l'autore del Plugin a impostare il nome desiderato e la funzione per tale pulsante, impostazione delle preferenze in [Plugin_somename file BTN](#).

L'approccio al Plugin "[Stuff](#)" è diverso in accordo con il tipo di persona:

Per l'utente finale, il giocatore di livello finale, l'unico file richiesto è la libreria [Plugin_somename.dll](#). solo che era nella cartella [TRLE](#) e tutto funzioni bene.

Diversamente, per i costruttori di livello, sarà necessario avere il file di installazione Plugin, installare il Plugin in [NG_Center](#) e per essere in grado di aggiungere comandi Script specifico (nel programma [NG_Center](#)) o dei suoi fattori scatenanti nel programma

[Mentre, per l'autore del plug-in, sarà necessario avere anche il codice sorgente per avviare lavoro. Se intendi creare un tuo Plugin, guarda il Software Development Kit per i Plugin.](#)

Nella pagina di cui sopra si trova il Link per scaricare il pacchetto "[Plugin_Sdk_Store](#)" e anche tutta la documentazione sui Plugin in costruzione.

A proposito di questo, ricordo che ho aggiunto molte strutture per ottenere un eventuale creazione del Plugin, anche per chi non è un vero e proprio programmatore, ma semplicemente un costruttore di livelli che vuole imparare qualcosa di nuovo.

I costruttori di livelli, con esperienza su Scripting avanzato, probabilmente saranno in grado di creare i propri Plugin, dal momento che in molti casi non è più complicato dello Scripting, in realtà, è molto spesso più facile, sicuramente si otterranno più obiettivi con poche righe di codice del Plugin di una lunga lista di comandi Script.

Per completare questo discorso, vi ricordo di [non costruire da zero un progetto C++ funzionante](#), poiché, nel pacchetto di [Plugin sdk](#), se avete già un progetto Plugin.

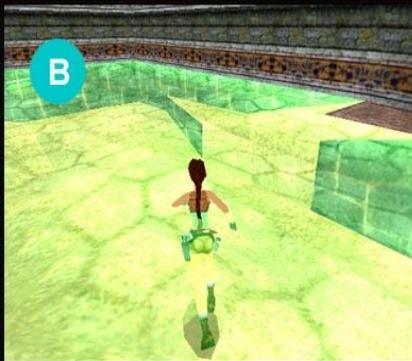
Dovrai solo aggiungere poche righe in modo giusto per creare i Trigger, oggetti o altre abilità.

NOTA: nel [Plugin sdk store](#), troverete che anche le fonti di tutti i Plugin utilizzati in questa demo e anche un [Tutorial di Html](#) per imparare, passo dopo passo, come creare questi nuovi oggetti e altre abilità.

FINE DEMO

Demo by Paolone traduzione in Italiano by Franky

ScreenShots



Descrizione

Sabbie Mobili (Quick-Sands)

[Download: QuickSand.zip \(2.5 Mb\)](#)

In questo esempio si dispongono di due diversi tipi di sabbie mobili: (Figura B) galleggianti/semitrasparenti, (Figura A) opache.

Informazioni sulle impostazioni: le sabbie mobili non richiedono alcuna impostazione nello [Script](#), ma è possibile impostare il movimento della superficie in

È possibile impostare il tipo di sabbia utilizzando la casella numerica bianca immediatamente a destra del pulsante speciale [\[Water/Rain/Snow/Q-sand\]](#) [Acqua/Pioggia/Neve/Sabbia] sotto il pannello 2d.

I primi tre valori "1", "2" e "3" operano come per l'acqua, con onde sempre più grandi aumentando il numero, mentre per sabbie mobili l'impostazione "4" forza una superficie piena (senza onde, come in (Figura A) , questo potrebbe essere utile per nascondere, alla vista del giocatore, le sabbie mobili, ottenendo un gioco più difficoltoso.

In questo progetto c'è anche un esempio per utilizzare il comando [Script GlobalTrigger](#) per imporre un Trigger (in questo caso ho usato un Trigger finale) quando Lara ha prelevato un determinato numero di oggetti (6 vasi canopi).

NOTA: questo esempio è necessario un [tombnextgeneration.dll](#) con la versione 1.1.8.4 o successiva, (20 agosto 2008)

FINE DEMO

ScreenShots

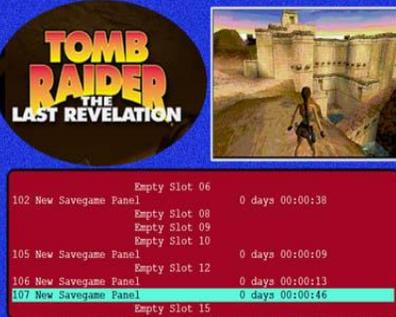
Load Game



Load Game



Load Game



Load Game



Descrizione

Pannello Savegame (Savegame Panel)

[Download: Savegame_Panel.zip \(8.8 Mb\)](#)

Dalla versione 1.2.0.8, il TRNG è in grado di supportare un nuovo pannello di **Savegame** per Caricare (**Load**) o Salvare (**Save**) il gioco.

Devi usare il comando nello Script **SavegamePanel=** per scegliere l'immagine di sfondo e il layout da utilizzare.

Con il nuovo pannello savegame si hanno diversi vantaggi:

1) Si può avere una **immagine per ogni savegame selezionato**, in questo modo per il giocatore sarà più facile cercare un savegame specifica.

2) Si può avere **più di 15 slot di savegame**, ma è possibile scegliere liberamente il numero di salvataggi gestibile in nello stesso momento (**Max 100 slot savegame**)

3) Per ogni savegame selezionato il giocatore può vedere tutte le informazioni su alcuni valori interni di quel savegame: **numero di medipack, armi disponibili, i segreti, la distanza percorsa** ect.

4) E' possibile creare l'immagine di **Background personalizzata** e scegliere tra **7** diversi layout, dove per "**layout**" si intende la disposizione delle diverse zone: lista savegame, immagine salvataggi e (opzionale) Info savegame.

Nella Figura a sinistra è possibile vedere alcune delle possibilità.

Se si sceglie un **layout** in cui le informazioni del savegame non sono presenti, è possibile utilizzare tale **spazio vuoto** per disegnare un **logo personalizzato** della vostra avventura o di qualche altro aspetto sorprendente.

FINE DEMO

Demo by Paolone traduzione in Italiano by Franky

ScreenShots



Descrizione

Modalità Standby (Stand-by Mode)

[Download: Standby.zip \(3 MB\)](#)

Questa demo mostra alcuni utilizzi del comando dello Script di [StandBy](#).

Lo [Standby](#) è utilizzato in modalità nativa, è una sorta di [auto-demo](#) che inizia nel gioco quando il giocatore non da i comandi nel gioco per qualche tempo.

L'**obiettivo principale di standby**, in modalità nativa, è quello di mostrare il volto di Lara, nel gioco si vede sempre di spalle, finalmente si potrà vedere la bellezza del suo voloto.

L'atteso nativo richiede solo un comando [Standby con ID = 1](#).

Se si inseriscono altri comandi [Standby](#) nella stessa sezione [\[Level\]](#), si dovranno digitare **ID diversi**, naturalmente. Questo [Standby](#) supplementare non sarà mai giocato da solo, ma è possibile attivarlo utilizzando la Flipeffect: "[Camera. Enable the](#) <&>[Standby camera effect for \(E\)seconds](#)"

Il motivo per utilizzare uno [Standby](#) tramite Flipeffect, potrebbe essere quella di avere una sorta di facile accesso alla [Telecamera Flyby](#). Per esempio, se si utilizza il tipo [TSB_PANORAMA](#), e si imposta una grande distanza di campo, è possibile mostrare una gran parte del vostro livello evitando, così, di creare una complicata sequenza di flyby.

I comandi di [Standby](#) che si vedono nello [Script.txt](#) fornito in questa Demo, sono solo esempi e si possono cambiare come si desidera. Ad esempio si potrebbe utilizzare un effetto [Matrix o Panorama](#) per l'attesa nativo ([ID=1](#)). Comunque in questo caso, ricordarsi di utilizzare il flag [FSB_DISABLE_ON_CRAMPED_SPACE](#), perché, in caso contrario, la modalità Standby potrebbe cominciare quando Lara è coperta da molti muri e il risultato in gioco sarebbe molto scarso.

Se non si desidera [disattivare](#) la modalità [Standby](#) in spazi angusti, una soluzione potrebbe essere quella di utilizzare una modalità di standby che non richiede grande spazio intorno a Lara.

Ad esempio un tipo di ritratto, con una breve distanza (minore di 1024) e il flag [FSB_IMMEDIATE](#), evita che la Telecamera si muova dalla posizione posteriore, fino a raggiungere la situazione [frontale](#) di Lara.

FINE DEMO

ScreenShots

Descrizione



Statics Demo (Statics Demo)

[Download: Statics.zip \(2.4 Mb\)](#)

Questo progetto richiede il TRNG con versione **1.1.8.7** o superiore.

In questa Demo ci sono esempi sui seguenti argomenti:

Collisione enorme

(Figura A) per gestire **Static** più grandi di **6x6 settori**.

Nel vecchio **tomb4**, uno **Static** **non può avere dimensioni maggiori di 6x6 settori**, se si è tentato di **creare Static più grande di 6x6** la collisione **non funziona bene**, e Lara è in grado di "entrare" da qualche lato di questi enormi **Static**.

Trasparenza Static

(Figura B) l'**OCB** imposta un livello di trasparenza per ogni **Static**, è possibile scegliere tra la **trasparenza del ghiaccio** o la **trasparenza del vetro**.

Colore Static

(Figura C) il **Trigger Flieffect** serve per aggiungere un effetto di colore allo **Static**, le modifiche del colore della colonna **Static** sono sul tono di colore.

Esplosione Static

(Figura D) mostra l'esplosione di uno **Static**, l'impostazione corretta dell'**OCB** permette di trasformare ogni **Static** in una bomba che esploderà quando Lara lo tocca, (Figura D), ecco cosa succede quando Lara tocca uno **Static** (in questo caso un vaso).

Static in Movimento

(Figura E) ora è possibile spostare uno **Static** all'interno della stessa stanza.

(Figura E) si vede un piccolo missile di colore rosso che si sta muovendo molto velocemente verso Lara uccidendola con una esplosione.

In questo Demo c'è anche un esempio di **Shatter Hard**, cioè uno **Static** che con un corretto **OCB** potrebbe diventare molto difficile da distruggere.

Lo **Shatter** può essere distrutto e va in frantumi, solo da **armi esplosive** e non da (pistole, revolver, normali munizione balestra).

NOTA: Anche da **Vehicles, Rollingballs e Sphinx** sono in grado di **distruggerlo**.

In questa **Demo** ci sono anche esempi su altri **OCB**, impostandoli possono essere in grado di avvelenare, bruciare o danneggiare Lara quando lo tocca.

Si tratta di un estratto della descrizione di **OCB Static** che potete trovare nel pannello di riferimento di **NG_Center**.

Da **TRNG** versione **1.1.8.7** è possibile impostare in ogni oggetto statico alcuni codici **OCB** per impostare le seguenti caratteristiche.

4 = disabilita la Collisione.

Lara potrebbe passare attraverso lo **Static**, (**nessuna collisione**).

8 = set vetro trasparenza.

Lo **Static** avrà un effetto di **trasparenza del vetro**.

16 = set ghiaccio trasparente.

Lo **Static** avrà una leggera **trasparenza**, come il **ghiaccio**.

32 = Danneggia Lara a contatto fisico.



Lo **Static** ferisce **Lara se lo tocca**, rimuove il danno 10 hp a Lara (versione vitalità = 1000) in ogni caso è possibile modificare questo valore utilizzando **CUST_SET_STATIC_DAMAGE** personalizzato con un comando **Script**.



64 = Ustiona Lara il contatto fisico.

Lara sarà bruciata e solo se troverà una pozza d'acqua si potrà salvare.

128 = A contatto fisico Lara esploderà.

Con questo **OCB** è possibile trasformare questo **Static** in una sorta di mina.

256 = A contatto fisico Lara sarà avvelenata.

Lara sarà avvelenata quando tocca lo **Static**, è possibile modificare l'intensità del veleno con il comando nello **Script CUST_SET_STATIC_DAMAGE**.

512 = Enorme collisione.

Informa il motore **TRNG** che il corrente **Static** ha una casella di **collisione più grande di settori 6x6**.

1024 = durezza Shatter.

Nella vecchio **tomb4** gli oggetti **Shatter** sono distrutti da tutte le altre armi, se si aggiunge **1024 a un OCB** di qualche oggetto di shatter, è possibile aumentare la durezza e lo **Static** sarà distrutto solo da munizioni esplosive (granata non flash), sphinks, Jeep, Sidecar e Rollingball (se ha un corretto **OCB**).

2048 = Trigger pesante sul contatto.

Quando Lara tocca uno **Static** con **OCB 2048** il **Trigger pesante** verrà attivato.

OSSERVAZIONI:

È possibile aggiungere i valori **OCB**, ma **non aggiungere mai due diversi tipi di trasparenza**.

Per esempio se si desidera usare il metodo di uno **Static** come il **vetro** e con nessuna collisione è possibile digitare il valore nella finestra **OCB=8+4=12**

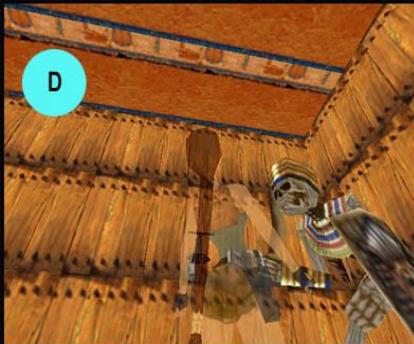
Altro esempio, per trasformare un **Static** in una **bomba** è stata in grado di far esplodere Lara e bruciarla mettere **OCB=64+128=192**

È possibile modificare in tempo di gioco questi attributi utilizzando il "metodo statico. ..." **Flipeffects**.

FINE DEMO

Demo by Paolone traduzione in Italiano by Franky

ScreenShots



Descrizione

Variabili TRNG (TRNG Variables)

[Download: TRNG Variables.zip \(3.2 Mb\)](#) (Stato il tutorial con la descrizione della [inventario critico zona di memoria](#)).

In questa [Demo](#) è possibile trovare anche un [Tutorial](#) sulle nuove [Variabili TRNG](#) e aree di memoria critiche per la creazione di nuove capacità.

(Figura A) nel progetto si possono trovare alcune dimostrazioni sull'utilizzo delle [variabili](#) e [memoria critica](#), potete vedere l'effetto creato modificando l'orientamento verticale (Fronte) di Lara.

(Figura B) la cosa più interessante in questa demo è il [super salto](#), ottenuto con l'aggiunta di un [jump-guadagno](#) quando Lara esegue una corsa prima di saltare.

Nel progetto sono state inserite anche altre novità come il nuovo interruttore di comando per selezionare diversi [TriggerGroup](#) in accordo con il valore di alcune variabili nello Script.

(Figura C) si può vedere una tastiera utilizzata per scegliere la porta da aprire e chiudere le altre nello stesso tempo.

(Figura D) un'altra caratteristica è il [congelamento del nemico](#), è possibile bloccare un nemico, che diventa come una statua, allora si può attivare con un altro Trigger aggiungendo anche alcuni effetti speciali come "vibrazione" o "Esplosione" per l'azione di risveglio.

FINE DEMO

ScreenShots



Descrizione

Trigger Verticali (Vertical Triggers)

[Download: Vertical Triggers.zip \(11 Kb\)](#)

Questo esempio mostra un possibile utilizzo del [Trigger Verticali](#).

(Figura A) i Trigger Verticali sono stati impostati per creare un muro scalabile non lineare, qui si può vedere il risultato finale, Lara è grado di salire solo dove si vedono le Texture a scala.

(Figura B) Se Lara cercherà di scavalcare i riquadri quadrate cadrà giù.

Il trucco per realizzare questo obiettivo consiste nel disattivare il tasto di [Azione](#) (necessario per mantenere lara nello stato di salita), quando l'altezza di Lara dal piano entra in qualche impostazione del [trigger verticale](#) nei settori chiusi per scalare la parete.

NOTA: è necessario costruire questo progetto utilizzando i file standard del livello [Catacombe](#) (WAD e TGA).

FINE DEMO