

Inhaltsverzeichnis

Vorwort.....	2
Installation.....	2
QSB-Datei.....	2
Die Mapskripte.....	2
DEBUG aktivieren.....	2
Aufträge im Skript erzeugen.....	3
Mission-Briefings und Cutscenes.....	3
Interaktive Objekte richtig anwenden.....	3
Nichtspieler-Charaktere nutzen.....	3

Vorwort

Die Symfonia-QSB, oder auch QSB Plus, hat das Ziel das Leben des Mappers zu erleichtern. Aber wie es stets der Fall ist mit neuen Technologien, muss auch hier ein klein wenig beachtet werden um in den Genuss der Vorzüge dieses Framework¹ zu kommen.

Nachfolgend werde ich die wichtigsten Dinge, angefangen mit der Installation, erklären.

Installation

QSB-Datei

Sobald du eine neue Map erstellt hast, muss die QSB in die Map *importiert* werden. Siehe dazu das SEED Handbuch, falls dir der Prozess nicht klar ist. Ein Release kommt mit *zwei* Versionen der QSB, die sich nur in einem Punkt unterscheiden: dem Speicherplatz.

Sobald du in den Release-Ordner gegangen bist, wirst Du zwei verschiedene Dateien vorfinden: `questsystembehavior.lua` und `questsystembehavior_min.lua`.

Die Datei `questsystembehavior.lua` ist die unveränderte QSB. Sie enthält noch alle Leerzeichen und Code-Kommentare und eignet sich zum stöbern, sollte Dir die Dokumentation nicht ausreichen um Deine Fragen zu klären.

Die Datei `questsystembehavior_min.lua` wurde auf ein minimum komprimiert, sodass sie etwa um die Hälfte weniger Speicherplatz benötigt. Allerdings ist der Code durch das Entfernen von Leerzeichen und Kommentaren sehr unleserlich.

Letztlich ist es egal, welche der beiden Dateien du importierst. Die minimierte QSB bietet dir nur den Vorteil weniger Speicher zu benötigen.

Die Mapskripte

Die QSB wird nicht mit den Standardskripten des Mapeditors funktionieren! Als erstes musst Du im Mapeditor den Expertenmodus aktivieren. Sobald das getan ist, kannst du eigene Skripte importieren. Siehe dazu das SEED Handbuch.

Im Release-Ordner befinden sich die Skripte `mapscript.lua` und `localmapscript.lua`. Kopiere Dir diese Dateien, denn mit ihnen wirst Du arbeiten.

DEBUG aktivieren

Du möchtest vorher vielleicht den DEBUG aktivieren, damit du Fehler schneller entdecken kannst. Füge dazu in der `Mission_FirstMapAction` im globalen Skript die Funktion `API.ActivateDebugMode` nach dem Aufruf von `Init InitKnightTitleTables` ein.

¹ Framework: ist ein Programmiergerüst, welches vereinfacht und dem Anwender Vorteile gegenüber der nativen Möglichkeiten einer Umgebung bietet.

Eine ausreichende Konfiguration ist folgende:

```
API.ActivateDebugMode(true, true, false, true);
```

Dies aktiviert alle Checks und die Konsole, aber nicht Quest Trace. Für mehr Informationen zum Debug konsultiere bitte die Dokumentation (<doc/index.html>).

Aufträge im Skript erzeugen

Mission-Briefings und Cutscenes

Interaktive Objekte richtig anwenden

Nichtspieler-Charaktere nutzen