

Version RE 1200

Dies war das initiales Release von Symfonia.

Datum der Veröffentlichung: 30.06.2018

Veröffentlicht unter: <https://drive.google.com/open?id=1z2US-7cGHkgd2kGyeFo096zzE6cjDn9U>

Inhalt des Archiv:

Datei	Beschreibung
qsb.lua	Enthält die gesamte QSB zusammen mit allen Kommentaren
qsb-min.lua	Enthält die komprimierte QSB (nur Funktionen)
mapscript.lua	Schablone für das globale Skript
localmapscript.lua	Schablone für das lokale Skript
CutsceneGuide.pdf	Anleitung zum Erstellen von „echten“ Cutscenes
GettingStarted.pdf	Einsteigeranleitung zu Symfonia
Start.txt	Informationen zum Archiv

Änderungen:

- **Cutscene Tool**

Die Skriptseitige Unterstützung von Jeans Tool für "echte" Cutscenes ist enthalten. Eine Anleitung wurde ebenfalls beigelegt.

- **Interaktive Objekte**

Kosten von interaktiven Objekten werden jetzt immer sofort bezahlt. Es fährt kein Karren mehr hin. Das senkt unnötige Wartezeiten. Belohnungen fahren weiterhin mit einem Karren vom Objekt zum Lagerhaus. Man muss also immer noch aufpassen, dass niemand den Wagen klaut! Zudem habe ich den Bug gefixt, dass kein Geologe zu einem leeren Brunnen kam.

- **Burglager**

Das Burglager hat aus technischen Gründen eine kleine Namensänderung erfahren:

BundleCastleStore -> AddOnCastleStore

Das hat jedoch keine Auswirkungen auf den Einsatz. Alle Funktionen sind weiterhin über den Alias QSB.CastleStore aufrufbar.

Der Grund dafür:

Das Burglager ist jetzt abhängig von den interaktiven Objekten. Dieser Schritt war nötig, damit Kosten von interaktiven Objekten auch direkt aus dem Burglager bezahlt werden können. Somit ist kein Auslagern mehr nötig!

- **Linksklicks in Briefings**

Es ist nun möglich in Briefings 3 verschiedene Positionen der Maus abzufragen, wenn der Spieler links klickt:

- Die Position der Maus auf dem Bildschirm
- Die Position der Maus auf der Spielwelt
- Das aktuelle Entity unter der Maus (das ist noch etwas buggy)

Dies geschieht über spezielle Funktionen innerhalb einer Briefing Page.

- **Flight-Funktionen für Fake Cutscenes**

Es gibt jetzt vereinfachte Funktionen für Fake-Cutscenes. Mit ihnen können Flights komfortabler notiert werden. Eine Anleitung ist dem Beginners Guide beigelegt. Altbewährte Schreibweisen funktionieren immer noch. (Aber irgendwann werde ich das glatt ziehen)

- **Log Levels**

Das Standard-Loglevel ist jetzt auf DEBUG gesetzt. Das bedeutet, dass Nachrichten weniger gewichtig als DEBUG nicht angezeigt werden. Diese sind dann nur im Log zu finden. Das Loglevel kann im Skript verändert werden. Siehe dafür die Dokumentation zum Debug.

- **Quest Behavior**

Einige Quest Behavior hatten fehlerhafte DEBUG-Methoden. Haben sie jetzt nicht mehr.

Zwischenversionen

Dies sind Versionen, die nicht öffentlich angekündigt wurden und ausschließlich Fehler im initialen Release beseitigten.

- Version RE 1205
- Version RE 1206
- Version RE 1300

Version RE 1400

Dies war der nächste Versionssprung, der einige Neuerungen mit sich brachte.

Datum der Veröffentlichung: 16.11.2018

Veröffentlicht unter: <https://drive.google.com/open?id=1z2US-7cGHkgd2kGyeFo096zzE6cjDn9U>

Inhalt des Archiv:

Datei	Beschreibung
qsb.lua	Enthält die gesamte QSB zusammen mit allen Kommentaren
qsb-min.lua	Enthält die komprimierte QSB (nur Funktionen)
mapscript.lua	Schablone für das globale Skript
localmapscript.lua	Schablone für das lokale Skript
CutsceneGuide.pdf	Anleitung zum Erstellen von „echten“ Cutscenes
GettingStarted.pdf	Einsteigeranleitung zu Symfonia
Start.txt	Informationen zum Archiv

Änderungen:

1. Debug

- Die Befehlszeile funktioniert jetzt getrennt von den Cheats.
- Quests werden nicht mehr zu dem Zeitpunkt getestet, zudem sie angelegt werden sondern nur noch zur Laufzeit. Das verhindert Probleme mit Abhängigkeiten zu anderen Quests, die es zum Zeitpunkt der Erzeugung nicht gibt.

- Parameter zum Aktivieren des Debug sind jetzt `_CheckAtRun`, `_QuestTrace`, `_DebugCheats`, `_DebugShell`
- Es ist jetzt möglich Kommandos zu verketten. Dadurch muss nicht mehrfach die Befehlszeile geöffnet werden.

Aus folgenden Eingaben

```
win QuestA
```

```
win QuestB
```

```
diplomacy 1 2 2
```

```
gexec CallMeMaybe()
```

wird nun

```
win QuestA & QuestB && diplomacy 1 2 2 && gexec CallMeMaybe()
```

(Einzelne Eingabe ist immer noch möglich.)

- Die Befehlszeilenkommandos sind jetzt an den internen Debugger angeschlossen. Mit der Funktion `eval()` kann man jedes beliebige Kommando ausführen.

```
eval("win QuestA & QuestB && diplomacy 1 2 2 && gexec  
CallMeMaybe()")
```

Hinweis: Der LuaDebugger funktioniert nur mit der LuaDebugger.dll. Die muss sich im bin-Verzeichnis befinden, sonst ist der Debugger nicht aktiv und man hat nur normales DSE.

2. Handelsfunktionen

- Einige Fehler in den Handelsfunktionen sind beseitigt.
- `API.RemoveOffer()` und `API.ModifyTradeOffer()` funktionieren gegenwärtig nicht.
- Die Diplomatieänderung des Schiffshändler kann abgeschaltet werden.
- Angebote durch `AddOffer()`, `AddMercenaryOffer()` und `AddEntertainerOffer()` können nur noch ein mal pro Lagerhaus vorkommen.

Version RE 1410

Nachdem es viel unnötiges Theater gab, wurde mit dieser Version eine Vergleichstabelle eingefügt.
Datum der Veröffentlichung: 04.12.2018

Veröffentlicht unter: <https://drive.google.com/open?id=1z2US-7cGHkgd2kGyeFo096zzE6cjDn9U>

Inhalt des Archiv:

Datei	Beschreibung
qsb.lua	Enthält die gesamte QSB zusammen mit allen Kommentaren
qsb-min.lua	Enthält die komprimierte QSB (nur Funktionen)
mapscript.lua	Schablone für das globale Skript
localmapscript.lua	Schablone für das lokale Skript
CutsceneGuide.pdf	Anleitung zum Erstellen von „echten“ Cutscenes
GettingStarted.pdf	Einsteigeranleitung zu Symfonia
VergleichQsbZweige.pdf	Vergleichstabelle QSB+ ↔ Symfonia
Start.txt	Informationen zum Archiv

Änderungen:

Bugfixes

- Fehler bei `API.RemoveTradeOffer()` behoben. Handelsangebote können jetzt einzeln von einem Lagerhaus entfernt werden.
- Fehler bei `API.ModifyTradeOffer()` behoben. Die Menge der Verfügbaren waren kann jetzt bestimmt werden.
- Der Fehler, der bei Kombination von Quest-Dialogen über `API.QuestDialog()` und Mapeditor Quests auftrat, ist behoben.
- (Fehler im Spiel) Parteien ohne Hauptgebäude haben jetzt ein Player Icon.

Features

- Player Portraits können jetzt komfortabel über `API.SetPlayerPortrait()` im globalen (und lokalen) Skript gesetzt werden (siehe Doku).
- Zu jedem Eintrag in einem Quest-Dialog kann nun ein Callback hinzugefügt werden.
- Quest-Dialoge warten nun in Echtzeit. Das gilt ausschließlich für Dialoge erzeugt mit `API.QuestDialog()`.

Verbesserungen

- Quellcode an einigen Stellen optimiert (hat keinen Effekt auf die Funktion).
- `Goal_InputDialog` speichert die Eingabe nun im zugehörigen Quest. Optional gibt es eine "freie Eingabe". Eingabe kann angefragt werden (siehe Doku).
- Regulärer Ausdruck für gültige Questnamen verbessert.
- Boolean-Umschrift nutzt nun einen Regulären Ausdruck anstelle von `if-elseif`.
- Außereißer in der Benennung von API-Funktionen wurden beseitigt (Konsistenz).

Version RE 1412

Datum der Veröffentlichung: ?

Veröffentlicht unter: <https://drive.google.com/open?id=1z2US-7cGHkgd2kGyeFo096zzE6cjDn9U>

Inhalt des Archiv:

Datei	Beschreibung
qsb.lua	Enthält die gesamte QSB zusammen mit allen Kommentaren
qsb-min.lua	Enthält die komprimierte QSB (nur Funktionen)
mapscript.lua	Schablone für das globale Skript
localmapscript.lua	Schablone für das lokale Skript
CutsceneGuide.pdf	Anleitung zum Erstellen von „echten“ Cutscenes
GettingStarted.pdf	Einsteigeranleitung zu Symfonia
VergleichQsbZweige.pdf	Vergleichstabelle QSB+ ↔ Symfonia
Versionshistorie.pdf	Abschrift der Release Notes aus dem Forum
Start.txt	Informationen zum Archiv

Änderungen:

1. Fehler in den Funktionen behoben:

- `API.WinAllQuests()`, `API.StopAllQuests()`, `API.StartAllQuests()`,
`API.RestartAllQuests()`, `API.FailAllQuests()`
- Falsch platzierte RdO-Basispreise und Ernerungsraten verschoben.
- Fliegender Händler kann nun auch der Reihe nach durch Angebote durchgehen. Die Funktion `API.TravelingSalesmanDiplomacyOverride()` kann das Verhalten jeder Zeit wieder ändern.
- Quest-Dialoge können nun direkt benannt werden. Der Name kann in Nachfolge-Quests für Trigger verwendet werden.
- `API.RestartQuestDialog()` kann genutzt werden um einen Dialog neu zu starten.
- `API.InterruptQuestDialog()` kann genutzt werden um einen Dialog anzuhalten.