SoundSets von Train Fever konvertieren

Inhaltsverzeichnis

- 1 Einleitung
- 2 Grundstruktur
- 3 tracks
- 4 events
- 5 updateFn
- 6 Vergleich

Alternativ zum Manuellen Konvertieren kann man bei aktivierter (transportfever.net/index.php/T…546-TF-Mod-Konvertierung/) Mod auch Train Fever Mods ohne Konvertierung öffnen

1 Einleitung

Wie angekündigt funktionieren die SoundSets aus Train Fever in Transport Fever nicht mehr, glücklicherweise ist es aber ohne großen Aufwand möglich sie zu konvertieren (solange man keine der neuen Funktionen nutzen möchte).

2 Grundstruktur

Als erstes muss ganz am Anfang der Datei folgende Zeile hinzugefügt werden:

```
local soundeffectsutil = require "soundeffectsutil"
```

Die weiteren Änderungen betreffen dann den Rückgabewert der data-Funktion, hier bleibt eigentlich nichts so wie es war, die benötigten Daten sind aber alle auch bereits in den alten SoundSets vorhanden. Die neue Grundstruktur sieht so aus:

Code

```
function                                                                    data()
return                                                                          {
 tracks                                    =                                     {
  {     name     =     "Pfad     zur     Datei",     refDist     =     25.0     }
 },
 events                                    =                                     {
  horn   =   {   names   =   {   "Pfad   zur   Datei"   },   refDist   =   16.0   },
 },
 updateFn                   =                   function                   (input)
  return                                                                          {
   tracks                                    =                                     {
    {         gain         =         1.0,         pitch         =         1.0         }
   },
   events                                    =                                     {
    {         gain         =         1.0,         pitch         =         1.0         }
   }
  }
 end
 }
end
```

Alles anzeigen

Der `type` fällt also komplett weg, und `refDist` wird an anderen Stellen benötigt.

3 tracks

Enthält alle Dateien die vorher unter `params` -> `tracks` -> `clip` vorhanden waren, also alles was in einer Schleife abgespielt werden soll. Die angesprochene `refDist` wird jetzt für jede Sounddatei einzeln angegeben.
Wenn ihr Bremssounds und das Quietschen der Räder nutzen wollt, müsst ihr noch diese Einträge hinzufügen:
Code

```
{ name = "environment/_no-license_train-wheels-ringing-curve.wav", refDist = 25.0 },
  { name = "vehicle/train_electric_modern/_brakes.wav", refDist = 25.0 },
```

4 events

Hier sind alle Sounds zu finden, die nur einmalig abgespielt werden, also z.B. Horn/Hupe, Türen öffnen/schließen oder das Klacken der Räder. Aus Einträgen wie `horn = "vehicle/ICE1/horn.wav"` in den alten Dateien wird hier `horn = { names = { "vehicle/ICE1/horn.wav" }, refDist = 16.0 }`. Bei den `names` sind auch mehrere Pfadangaben möglich, wie genau das Spiel damit umgeht kann ich aber gerade nicht mit Sicherheit sagen (wird z.B. bei den `clacks` verwendet).

5 updateFn

Hier wird entschieden welcher `track` bzw. welches `event` unter welchen Bedingungen wie laut/schnell abgespielt wird, die Funktion ersetzt also die `gainCurve` und `pitchCurve`. Zum Konvertieren könnt ihr als wert einfach die Funktion `soundeffectsutil.sampleCurve()` verwenden und als Argumente die Nodes aus der alten Datei und `input.speed01` übergeben, also z.B:

Code

```
--                                                                                    alt
gainCurve                                        =                                      {
 nodes                                          =                                        {
  {                                        .0,                                        0.6},
  {                        .01,                          0.12                            },
  {                        .05,                           0.0                            },
 }
},


--                                                                                    neu
gain                        =                        soundeffectsutil.sampleCurve({
 {                                        .0,                                        0.6},
 {                        .01,                          0.12                            },
 {                        .05,                           0.0                            },
}, input.speed01),
```

Alles anzeigen

Die Einträge hier beziehen sich immer auf die jeweiligen Einträge am Anfang der Datei, es müssen in den Tabellen also auch immer die gleiche Zahl an Einträgen vorhanden sein.

Bremssounds und Quietschen der Räder:

Code

```
soundeffectsutil.squeal(input.speed,    input.sideForce,    input.maxSideForce),
    soundeffectsutil.brake(input.speed, input.brakeDecel, 0.5),
```

Für Events wie das Horn können einfach, wie in der Grunstruktur angegeben, feste Werte verwendet werden. Die `clacks` verwenden im Originalspiel die Funktion `soundeffectsutil.clacks(input.speed, input.weight, input.numAxles, axleRefWeight, input.gameSpeedUp)`, wobei `axleRefWeight` die Achslast ist und entweder vorher definiert oder direkt als Zahl angegeben werden muss.
Zusätzlich zur Geschwindigkeit stehen über `input` jetzt noch mehr Werte zur Verfügung die zur Bestimmung der gain/pitch-Kurven genutzt werden können, z.B. `power01` für die abgerufene Leistung.

6 Vergleich

Anhand des SoundSets des ICE 1:
Code

```
function data()
return {
 type = "GENERIC",
 refDist = 16.0,
 horn = "vehicle/ICE1/horn.wav",
 params = {
  tracks = {
   {
    clip = "vehicle/ICE1/idle.wav",
    gainCurve = {
     nodes = {

      {.01, 0.12 },
      {.05, 0.0 },
     }
    },
    pitchCurve = {
     nodes = {
      {.0, 2},
      {.01, 2},
     }
    }
   },
   {
    clip = "vehicle/ICE1/drive.wav",
    gainCurve = {
     nodes = {
      {0.0, 0.0 },
      {0.01, 1.0 },
      {1.0, 2.0 },
     }
    },
    pitchCurve = {
     nodes = {
      {0, 1.0 },
      {0.01, 1.0 },
      {1, 1.5},
     }
    }
   },
  },
 }
}
end
```

Alles anzeigen

Code

```
local soundeffectsutil = require "soundeffectsutil"
function data()
return {
 tracks = {
  {     name     =     "vehicle/ICE1/idle.wav",     refDist     =     25.0     },
  {     name     =     "vehicle/ICE1/drive.wav",     refDist     =     25.0     },
  { name = "environment/_no-license_train-wheels-ringing-curve.wav", refDist = 25.0 },
  {   name   =   "vehicle/train_electric_modern/_brakes.wav",   refDist   =   25.0   }
 },
 events = {
  clacks = {
   names = {
    "vehicle/clack/modern/part_1.wav",
    "vehicle/clack/modern/part_2.wav",
    "vehicle/clack/modern/part_3.wav",
    "vehicle/clack/modern/part_4.wav",
    "vehicle/clack/modern/part_5.wav",
    "vehicle/clack/modern/part_6.wav",
    "vehicle/clack/modern/part_7.wav",
    "vehicle/clack/modern/part_8.wav",
    "vehicle/clack/modern/part_9.wav",
    "vehicle/clack/modern/part_10.wav"
   },
   refDist = 25.0
  },
  horn   =   {   names   =   {   "vehicle/ICE1/horn.wav"   },   refDist   =   16.0   }
 },


 updateFn = function (input)
  local axleRefWeight = 10.0
  return {
   tracks = {
    {
     gain = soundeffectsutil.sampleCurve({

      {  .01,  0.12  },
      {  .05,  0.0  },
     }, input.speed01),
     pitch = soundeffectsutil.sampleCurve({
      {  .0,  2},
      {  .01,  2},
     }, input.speed01)
    },
    {
     gain = soundeffectsutil.sampleCurve({
      {  0.0,  0.0  },
      {  0.01,  1.0  },
      {  1.0,  2.0  },
     }, input.speed01),
     pitch = soundeffectsutil.sampleCurve({
      {  0,  1.0  },
      {  0.01,  1.0  },
      {  1,  1.5},
     }, input.speed01)
    },
    soundeffectsutil.squeal(input.speed,    input.sideForce,    input.maxSideForce),
    soundeffectsutil.brake(input.speed,    input.brakeDecel,    0.5)
   },
   events = {
    soundeffectsutil.clacks(input.speed, input.weight, input.numAxles, axleRefWeight, input.gameSpeedUp),
    horn      =      {      gain      =      1.0,      pitch      =      1.0      }
   }
  }
 end
 }
end
```


https://dev.transportfever.net/lexicon/entry/152-soundsets-von-train-fever-konvertieren/

Alles anzeigen