

# Tutorial LS19 Modden

## Mods von IC auf SIC umbauen

### 1. Vorbereitung:

Man braucht dazu den GIANTS-Editor und einen Text-Editor (z.B NotePad++), sowie den Mod „SimpleIC“ von Modelleicher und einen „playerTrigger“.

Der Mod „SimpleIC“ kommt in den „mods-Ordner“.

Den „playerTrigger“ findet man im Daten-Anhang. Er muss an den Mod angepasst, sprich skaliert werden.

### 2. Arbeiten im GIANTS-Editor:

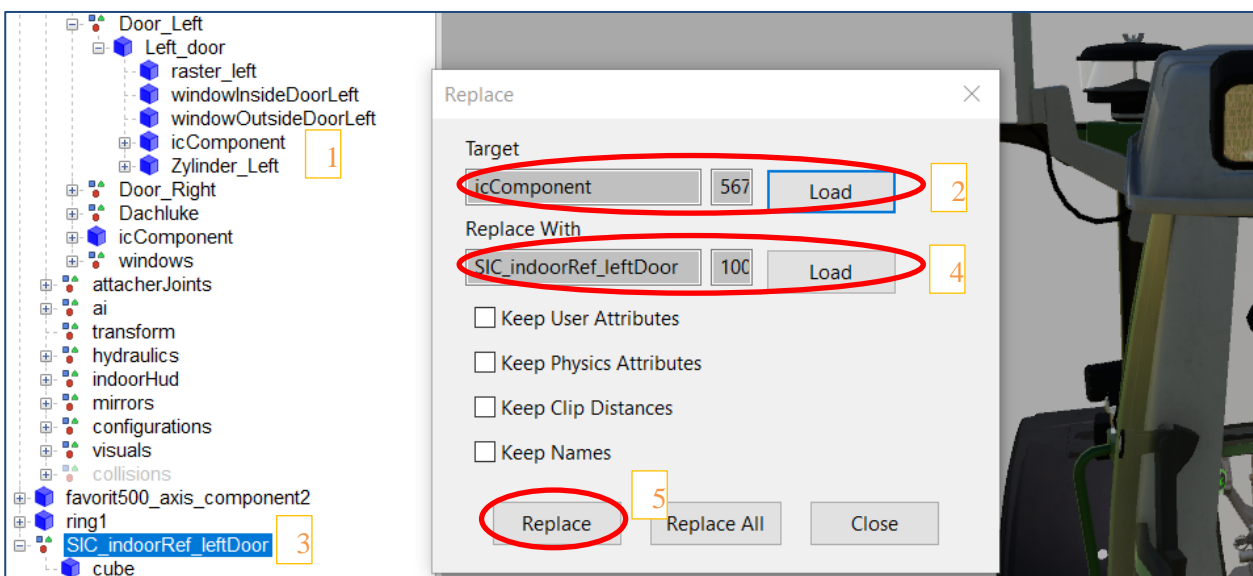
Den umzubauenden Mod im GE öffnen. Ich erkläre hier den Umbau am Beispiel der linken Tür.

Das erste was man macht, ist eine neue TG erstellen, die benennt man "SIC\_indoorRef\_leftDoor" (wenn die Tür von innen, also von der IndoorCamera aus bedient werden soll).

Bei den Türen kann man auch von außen bedienen, also von der OutdoorCamera. Dann hieße die TG "SIC\_outDoorRef\_leftDoor".

Als nächstes erstellt man sich einen Cube (**Create>Primitives>Cube**). Den skaliert man auf 0.05 0.05 0.05 und schiebt ihn in die zuvor erstellte TG.

Jetzt sucht man den IC-Marker der linken Tür und markiert ihn, aktiviert die Funktion **Edit>Replace**. Man wird feststellen, dass der zuvor markierte Marker links oben drinsteht. Rechts daneben klickst Du auf "load". Dann markiert man die neue TG "SIC\_indoorRef...") In der zweiten Zeile des Replace-Fenster links steht die dann drin und man lädt sie mit "load"



Dann auf "Replace". Somit hat man den IC-Marker gegen den SIC-Ref ausgetauscht und er ist an derselben Position wie vorher der IC-Marker.

Das macht man bei allen Markern so, man muss auch nicht jedes Mal eine neue TG erstellen, denn die erste wird eigentlich nur kopiert. (Umbenennen muss man natürlich schon, zu

Übersichtlichkeit) Der Cube dient nur zur Kontrolle, damit man sieht, wo genau das Kreuz später INGAME auftaucht. Dieser Cube wird am Ende des Umbaus, wenn alles passt, gelöscht.

Wenn man die Türen auch von außen öffnen will, muss eine zusätzliche TG manuell hinzugefügt und z.B. ans Türschloss geschoben werden. Jetzt merkt man auch wofür der "Cube" eigentlich gut ist (aber nicht nur den Cube verschieben, die ganze TG !!!). Ich weiß warum ich das hier schreibe, das passiert mir auch noch ab und zu.

Wenn das alles erledigt ist, muss man den "PlayerTrigger" aus dem Dateien-Anhang importieren und in das erste Haupt-Mesh schieben. Damit wären die Arbeiten im GE erstmal fertig. Speichern und schließen.

### 3. Arbeiten in der „vehicle-XML“:

Die folgenden Einträge unter den Bereich „<animations> </animations>“ einfügen. Sie ersetzen den Teil „<interactiveComponents> </ interactiveComponents >“.

```
</animations>
<simpleIC outsideInteractionTrigger="playerTrigger" >
  <animation animationName="door_left" animationSpeed="1" sharedAnimation="false" soundVolumeIncreasePercentage="0.5" >
    <insideTrigger triggerPoint="SIC_doorLeft_insideRef" triggerPointSize="0.04" />
    <outsideTrigger triggerPoint="SIC_doorLeft_outsideRef" triggerPointSize="0.04" />
  </animation>
```

An diesem Beispiel für die linke Tür registriert man die neuen SIC\_Refs:

#### Erklärung:

**animationName** -- muss identisch sein mit der Animation aus dem Bereich "<animations>"

**soundVolumeIncreasePercentage="0.5"** -- dadurch erhöht sich der indoorSound bei ausgeführter Animation, hier durch die geöffnete Tür hört man den Motor lauter.

Soll sich der IndoorSound nicht erhöhen z.B. bei klappbaren Sonnenblendenden, einblendbaren Terminals etc., den Wert verändern auf "0.0"

**insideTrigger** bzw. **outsideTrigger** -- erklären sich von selbst. Die blauen Einträge muss man im „i3dMapping“ registrieren (Wie das geht, sollte man wissen, alternativ kann man auch einfach den Index des entsprechenden Refs eintragen)

```
<!-- SIC_RefPoints -->
<i3dMapping id="playerTrigger" node="0>14"/>
<i3dMapping id="SIC_doorLeft_insideRef" node="0>4|13|0|3"/>
<i3dMapping id="SIC_doorLeft_outsideRef" node="0>4|13|0|5"/>
```

Bitte nicht vergessen, den SimpleIC-Bereich mit </simpleIC> abzuschließen und natürlich speichern.

### 4. Arbeiten in der „modDesc-XML“:

Man entfernt die "spezializations "interactivControl" und "interativWindows" aus den Bereichen "vehicleTypes" und "spezializations"

```
<vehicleTypes>
  <type name="XXXXXXX" parent="baseDrivable" className="Vehicle" filename="$data5/scripts/vehicles/Vehicle.lua">
    <specialization name="interactiveControl" />
    <specialization name="interactiveWindows" />
    <specialization name="addConfig" />
    <specialization name="extendedAnimationSounds" />
  </type>
</vehicleTypes>

<specializations>
  <specialization name="interactiveControl" className="InteractiveControl" filename="Scripts/InteractiveControl.lua" />
  <specialization name="interactiveWindows" className="InteractiveWindows" filename="Scripts/InteractiveWindows.lua" />
  <specialization name="extendedAnimationSounds" className="ExtendedAnimationSounds" filename="Scripts/ExtendedAnimationSounds.lua" />
  <specialization name="addConfig" className="AddConfig" filename="Scripts/AddConfig.lua" />
</specializations>
```

Ebenso den gesamten Bereich "actions" und "inputBinding" und in Bereich >110n> IC und animations

## 5. Arbeiten in den Ordnern:

Aus dem Ordner "Scripts" löscht man die 3 interactiv.lua  
Aus dem Ordner „Textures“ löscht man die „marker-Textur“.

## 6. Erster Check:

Nachdem man die vorherigen Arbeiten sorgfältig durchgeführt hat, packt man den Mod zu einer Zip-Datei und schiebt ihn in den „mods-Ordner“, startet das Spiel, aktiviert den Mod und kauft ihn.

Evtl. wird man feststellen, dass die IndoorCamera nicht alle SIC-Refs erreicht, so dass sie sich nicht „rot“ färben. Die muss man im GE abschließend noch anpassen, verschieben und wenn alles passt die "Cubes“ löschen oder unsichtbar machen.

Das war es! Ich wünsche allen viel Erfolg mit diesem Tutorial.

Euer Remraf