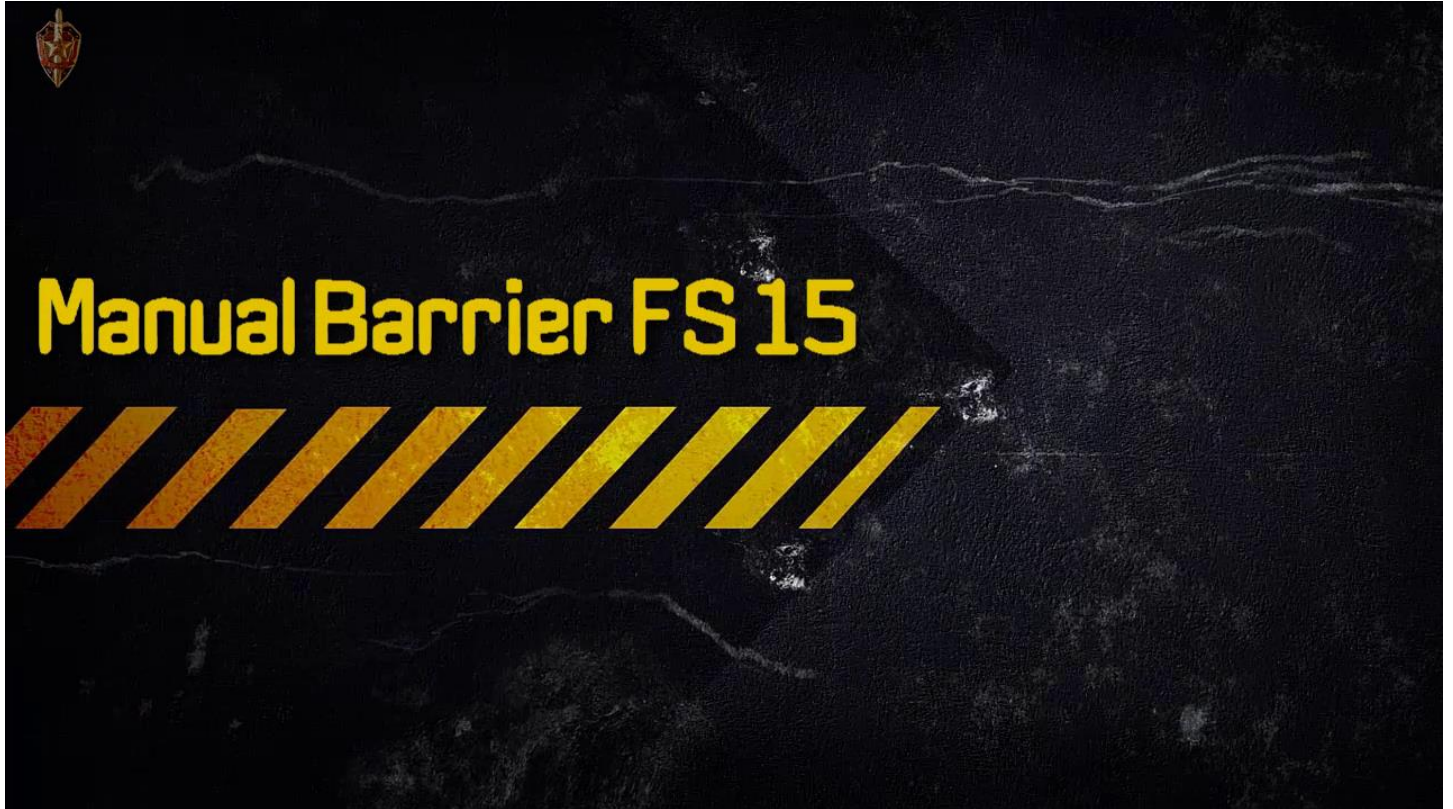


# **MANUAL BARRIER**

für den Landwirtschafts Simulator 2015



## **Inhaltsverzeichnis**

2	Allgemeine Beschreibung
3	Erklärung der User Attribute
7	Erklärung des Aufbaus im Scenograph
8	Erklärung der modDesc.xml Einträge
10	Tipps

## **Allgemeine Beschreibung**

ManualBarrier ist für Karten-Ersteller gedacht die viele unterschiedliche Objekte auf ihrer Karte entweder manuell oder automatisch in Bewegung versetzen wollen.

Eigentlich täuscht der Name denn dieses Paket kann weitaus mehr als nur eine Schranke manuell zu bedienen.

Egal ob nun ein Schiebetor, eine Schwingtür, eine Schranke oder gar ein komplex animiertes Objekt in Bewegung versetzt werden muss, manuell oder automatisch spielt dabei keine Rolle, oder ob es sich um einschaltbare Lichtquellen oder sonstiges handelt. Für all dies kann ManualBarrier verwendet werden.

Im Grunde kann ManualBarrier für so gut wie jedes Objekt verwendet werden, naja auf jeden Fall für die meisten ☺

Und dabei ist es sehr vielfältig, alles kann miteinander kombiniert werden, ob Rotation und Bewegung oder Animation und Licht, oder auch alles zusammen, ob man nun eine alte Leuchtstoffröhre mit dem typischen einschaltflackern haben möchte oder ein Rundumlicht bei Schiebetoröffnung. Man will den Zugang zu einem Ort nur zu einer bestimmten Uhrzeit gewähren? Kein Problem, auch das kann ManualBarrier. Automatisches Hoflicht ab 20 Uhr? Warum nicht, ManualBarrier macht auch dies einfach.

Selbstverständlich ist ManualBarrier auch für den Multiplayer-Modus von LS15 geeignet, jede Funktion wird dabei unterstützt.

Für den genauen Einbau folgen eine Erklärung der User Attribute, ein Beispielaufbau im I3D-Editor Scenegraphen sowie die nötigen Einträge in der modDesc.xml.

Grundsätzlich setze ich jedoch Kenntnisse mit dem Giants-Editor sowie der XML-Bearbeitung voraus damit ManualBarrier Verwendung finden kann.

## Erklärung der User Attribute

### animationClip

- Werte-Typ: String
- Standardwert: nicht vergeben
- Verarbeitungsvoraussetzung: **typeAnimated** = TRUE
- gibt den Namen des Clips an welcher für die Animation festgelegt wurde
- wurde der Name falsch angegeben wird keine Animation aufgerufen

### animatorIndex

- Werte-Typ: String
- Standardwert: nicht vergeben
- Verarbeitungsvoraussetzung: **typeAnimated** = TRUE
- gibt den Index zu einem animierten Objekt an
- sind mehrere Objekte in einem **animationClip** animiert ist es egal welcher von den Objekten über diesen Index angesprochen wird

### audioIndex

- Werte-Typ: String
- Standardwert: nicht vergeben
- gibt den Index zu einer AudioSource an welche bei der ausgeführten Bewegung abgespielt werden soll
- wenn angegeben wird der AudioClip abgespielt sobald sich das Objekt (Barrier|Gate|AnimatedObject) bewegt

### automaticClose

- Werte-Typ: Float
- Standardwert: 18
- Wertebereich: 0 bis 23.9999
- Verarbeitungsvoraussetzung: **automaticMode** = TRUE
- gibt die Uhrzeit an zu der das Objekt den geschlossenen Zustand annehmen soll

### automaticMode

- Werte-Type: Boolean
- Standardwert: FALSE
- dient zum Setzen von automatischen Öffnungs- sowie Schließzeiten
- wenn Objekt NICHT auf manuelle Öffnung eingestellt ist wird das Objekt automatisch zur **automaticOpen** Uhrzeit geöffnet und verbleibt bis zur **automaticClose** Uhrzeit im offenen Zustand
- ist das Objekt auf manuelle Öffnung eingestellt so gelten folgende Regeln (Halbautomatischer Betrieb)
  - + das Objekt kann zu jeder Zeit geöffnet werden
  - + das Objekt wird automatisch den geschlossenen Zustand annehmen wenn die **automaticClose** Uhrzeit erreicht ist
  - + wird das Objekt während in der Zeit zwischen **automaticClose** und **automaticOpen** geöffnet und der Trigger im Anschluss wieder verlassen so schließt sich das Objekt selbstständig

**User Attributes**

Name: gate

**Attributes**

Attribute	Value	Icon
animationClip	rollgate_rul	X
animatorIndex	0>2 0	X
audioIndex	0>3	X
automaticClose	18	X
automaticMode	<input type="checkbox"/>	X
automaticOpen	7	X
automaticStrict	<input type="checkbox"/>	X
flickerMode	<input type="checkbox"/>	X
lightIndex	0>2 0 0	X
lightOnMove	<input type="checkbox"/>	X
manualBarrierId	1	X
manualOpen	<input type="checkbox"/>	X
maxX	0	X
maxXAngle	0	X
maxY	0	X
maxYAngle	0	X
maxZ	0	X
maxZAngle	0	X
minX	0	X
minXAngle	0	X
minY	0	X
minYAngle	0	X
minZ	0	X
minZAngle	0	X
onCreate	modOnCreate.ManualBarrier	X
speedScale	60	X
string_Close	string_CLOSE	X
string_Name	string_DEFAULT	X
string_Open	string_OPEN	X
triggerIndex	0>0	X
typeAnimated	<input type="checkbox"/>	X
typeBarrier	<input type="checkbox"/>	X
typeGate	<input type="checkbox"/>	X
typeLight	<input type="checkbox"/>	X

## automaticOpen

- Werte-Typ: Float
- Standardwert: 7
- Wertebereich: 0 bis 23.9999
- Verarbeitungsvoraussetzung: **automaticMode** = TRUE
- gibt die Uhrzeit an zu der das Objekt den geöffneten Zustand annehmen soll

## automaticStrict

- Werte-Typ: Boolean
- Standardwert: FALSE
- Verarbeitungsvoraussetzung: **automaticMode** = TRUE  
**manualOpen** = TRUE
- greift nur wenn das Objekt mit manueller Öffnung versehen ist
- es gelten ähnliche Regeln wie beim Halbautomatischem Modus (siehe **automaticMode**) abgesehen von folgenden Änderungen
  - + ist die **automaticClose** Uhrzeit erreicht wird sich das Objekt schließen und kann bis zum eintreten der **automaticOpen** Uhrzeit NICHT wieder geöffnet werden

## flickerMode

- Werte-Typ: Boolean
- Standardwert: FALSE
- Verarbeitungsvoraussetzung: **typeLight** = TRUE
- das unter **lightIndex** angegebene Licht wird mit einem flackern (ähnlich einer alten Leuchtstoffröhre) gestartet

## lightIndex

- Werte-Typ: String
- Standardwert: nicht vergeben
- gibt den Index zu einem Objekt an welches eingeblendet werden soll

## lightOnMove

- Werte-Typ: Boolean
- Standardwert: FALSE
- gibt an ob ein Objekt auch dann noch eingeblendet bleiben soll wenn sich das Objekt nach Verlassen des Triggers schließt
- ist dieser Wert nicht angegeben (= FALSE) dann wird das unter **lightIndex** angegebene Objekt nach Verlassen des Triggers direkt ausgeblendet, unabhängig von der Bewegung anderer Objekte des Triggers

## manualBarrierId

- Werte-Typ: Integer
- Standardwert: 1 ( !!!)
- Wertebereich: 1 bis 65535
- gibt die fortlaufende Nummer zu den mit Manual Barrier Scripten gesteuerten Objekten an
- darf NICHT mehrfach verwendet werden da andernfalls lediglich das letzte Objekt welches die gleiche ID hat korrekt arbeiten kann

**User Attributes**

Name: gate

**Attributes**

Attribute	Value	Checkbox
animationClip	rollgate_rul	<input checked="" type="checkbox"/>
animatorIndex	0>2 0	<input checked="" type="checkbox"/>
audioIndex	0>3	<input checked="" type="checkbox"/>
automaticClose	18	<input checked="" type="checkbox"/>
automaticMode	<input type="checkbox"/>	<input checked="" type="checkbox"/>
automaticOpen	7	<input checked="" type="checkbox"/>
automaticStrict	<input type="checkbox"/>	<input checked="" type="checkbox"/>
flickerMode	<input type="checkbox"/>	<input checked="" type="checkbox"/>
lightIndex	0>2 0 0	<input checked="" type="checkbox"/>
lightOnMove	<input type="checkbox"/>	<input checked="" type="checkbox"/>
manualBarrierId	1	<input checked="" type="checkbox"/>
manualOpen	<input type="checkbox"/>	<input checked="" type="checkbox"/>
maxX	0	<input checked="" type="checkbox"/>
maxXAngle	0	<input checked="" type="checkbox"/>
maxY	0	<input checked="" type="checkbox"/>
maxYAngle	0	<input checked="" type="checkbox"/>
maxZ	0	<input checked="" type="checkbox"/>
maxZAngle	0	<input checked="" type="checkbox"/>
minX	0	<input checked="" type="checkbox"/>
minXAngle	0	<input checked="" type="checkbox"/>
minY	0	<input checked="" type="checkbox"/>
minYAngle	0	<input checked="" type="checkbox"/>
minZ	0	<input checked="" type="checkbox"/>
minZAngle	0	<input checked="" type="checkbox"/>
onCreate	modOnCreate.ManualBarrier	<input checked="" type="checkbox"/>
speedScale	60	<input checked="" type="checkbox"/>
string_Close	string_CLOSE	<input checked="" type="checkbox"/>
string_Name	string_DEFAULT	<input checked="" type="checkbox"/>
string_Open	string_OPEN	<input checked="" type="checkbox"/>
triggerIndex	0>0	<input checked="" type="checkbox"/>
typeAnimated	<input type="checkbox"/>	<input checked="" type="checkbox"/>
typeBarrier	<input type="checkbox"/>	<input checked="" type="checkbox"/>
typeGate	<input type="checkbox"/>	<input checked="" type="checkbox"/>
typeLight	<input type="checkbox"/>	<input checked="" type="checkbox"/>

## manualOpen

- Werte-Typ: Boolean
- Standardwert: FALSE
- gibt an dass das Objekt manuell zu steuern ist
- ist der Wert TRUE lässt sich das Objekt komplett manuell öffnen/anschalten und schließen/abschalten

## maxX, minX, maxY, minY, maxZ, minZ

- Werte-Type: Float
- Standardwert: 0.0
- Wertebereich: -65535 bis 65535
- Verarbeitungsvoraussetzung: **typeGate** = TRUE
- **min#** gibt die Ausgangsposition für zu verschiebende Objekte an, definieren gleichzeitig die geschlossene Position
- **max#** gibt die maximale Offen-Position an
- es müssen nicht alle Achsen angegeben werden, lediglich die Achse welche bewegt werden soll, andere nicht angegebene Achsen verbleiben in ihrer Ausgangsposition
- wenn mehrere Objekte zu verschieben sind müssen diese die gleichen Ausgangswerte besitzen

## maxXAngle, minXAngle

## maxYAngle, minYAngle

## maxZAngle, minZAngle

- Werte-Typ: Float
- Standardwert: 0.0
- Wertebereich: -65535 bis 65535
- Verarbeitungsvoraussetzung: **typeBarrier** = TRUE
- **min#Angle** gibt die minimale Rotation in Grad der entsprechenden Achse an, definiert gleichzeitig die geschlossene Position
- **max#Angle** gibt die maximale Offen-Rotation in Grad an
- es müssen nicht alle Achsen angegeben werden, lediglich die Achse welche rotiert werden soll, andere nicht angegebene Achsen verbleiben in ihrer Ausgangsrotation
- wenn mehrere Objekte zu rotieren sind müssen diese die gleichen ausgangswerte besitzen

## onCreate

- Werte-Typ: ScriptCallback
- Standardwert: nicht vergeben, für diese Funktion notwendig ist **modOnCreate.ManualBarrier**
- legt den Aufruf für das Script fest
- ohne diesen Aufruf wird sich das Objekt nicht bewegen oder auf Trigger-Kollisionen reagieren

## speedScale

- Werte-Typ: Float
- Standardwert: 20
- Wertebereich: 1 bis 65535
- legt die Geschwindigkeit der Bewegung/Rotation/Animation fest
- niedrige Werte verlangsamen, höhere Werte beschleunigen die Bewegung/Rotation/Animation

User Attributes	
Name	gate
animationClip	rollgate_rul
animatorIndex	0>2 0
audioIndex	0>3
automaticClose	18
automaticMode	<input type="checkbox"/>
automaticOpen	7
automaticStrict	<input type="checkbox"/>
flickerMode	<input type="checkbox"/>
lightIndex	0>2 0 0
lightOnMove	<input type="checkbox"/>
manualBarrierId	1
manualOpen	<input type="checkbox"/>
maxX	0
maxXAngle	0
maxY	0
maxYAngle	0
maxZ	0
maxZAngle	0
minX	0
minXAngle	0
minY	0
minYAngle	0
minZ	0
minZAngle	0
onCreate	modOnCreate.ManualBarrier
speedScale	60
string_Close	string_CLOSE
string_Name	string_DEFAULT
string_Open	string_OPEN
triggerIndex	0>0
typeAnimated	<input type="checkbox"/>
typeBarrier	<input type="checkbox"/>
typeGate	<input type="checkbox"/>
typeLight	<input type="checkbox"/>

## string\_Close

- Werte-Typ: String
- Standardwert: string\_CLOSE (= schließen)
- setzt die Textvariable für den Schließvorgang für die Anzeige im Hilfefenster des Spiels
- Werte werden in der **modDesc.xml** der Karte definiert

## string\_Name

- Werte-Typ: String
- Standardwert: string\_DEFAULT (= Tor/Schranke)
- setzt die Textvariable für den Namen des Objektes für die Anzeige im Hilfefenster des Spiels
- Werte werden in der **modDesc.xml** der Karte definiert

## string\_Open

- Werte-Typ: String
- Standardwert: string\_OPEN (= öffnen)
- setzt die Textvariable für den Öffnungsvorgang für die Anzeige im Hilfefenster des Spiels
- Werte werden in der **modDesc.xml** der Karte definiert

## triggerIndex

- Werte-Typ: String
- Standardwert: nicht vergeben
- gibt den Index zum Trigger an
- wird kein Trigger angegeben kann das Objekt nur funktionieren wenn **manualOpen** = FALSE und **automaticMode** = TRUE ist, andernfalls wird das Objekt keine Funktion darstellen

## typeAnimated

- Werte-Typ: Boolean
- Standardwert: FALSE
- gibt an ob das Objekt ein animiertes Objekt ist und demnach eine Animation enthält
- kann mit allen anderen Typen kombiniert werden
- wird deaktiviert wenn keine Animation zugewiesen und/oder keine animierter Index zugewiesen wird

## typeBarrier

- Werte-Typ: Boolean
- Standardwert: FALSE
- gibt an ob das Objekt ein zu rotierendes Objekt ist
- kann mit allen anderen Typen kombiniert werden

## typeGate

- Werte-Typ: Boolean
- Standardwert: FALSE
- gibt an ob das Objekt ein zu bewegendes (verschiebendes) Objekt ist
- kann mit allen anderen Typen kombiniert werden

## typeLight

- Werte-Typ: Boolean
- Standardwert: FALSE
- gibt an ob das Objekt ein/ausgeblendet werden soll
- kann mit allen anderen Typen kombiniert werden

**User Attributes**

Name: gate

Attributes	Value	Enabled
animationClip	rollgate_rul	<input checked="" type="checkbox"/>
animatorIndex	0>2 0	<input checked="" type="checkbox"/>
audioIndex	0>3	<input checked="" type="checkbox"/>
automaticClose	18	<input checked="" type="checkbox"/>
automaticMode	<input type="checkbox"/>	<input checked="" type="checkbox"/>
automaticOpen	7	<input checked="" type="checkbox"/>
automaticStrict	<input type="checkbox"/>	<input checked="" type="checkbox"/>
flickerMode	<input type="checkbox"/>	<input checked="" type="checkbox"/>
lightIndex	0>2 0 0	<input checked="" type="checkbox"/>
lightOnMove	<input type="checkbox"/>	<input checked="" type="checkbox"/>
manualBarrierId	1	<input checked="" type="checkbox"/>
manualOpen	<input type="checkbox"/>	<input checked="" type="checkbox"/>
maxX	0	<input checked="" type="checkbox"/>
maxXAngle	0	<input checked="" type="checkbox"/>
maxY	0	<input checked="" type="checkbox"/>
maxYAngle	0	<input checked="" type="checkbox"/>
maxZ	0	<input checked="" type="checkbox"/>
maxZAngle	0	<input checked="" type="checkbox"/>
minX	0	<input checked="" type="checkbox"/>
minXAngle	0	<input checked="" type="checkbox"/>
minY	0	<input checked="" type="checkbox"/>
minYAngle	0	<input checked="" type="checkbox"/>
minZ	0	<input checked="" type="checkbox"/>
minZAngle	0	<input checked="" type="checkbox"/>
onCreate	modOnCreate.ManualBarrier	<input checked="" type="checkbox"/>
speedScale	60	<input checked="" type="checkbox"/>
string_Close	string_CLOSE	<input checked="" type="checkbox"/>
string_Name	string_DEFAULT	<input checked="" type="checkbox"/>
string_Open	string_OPEN	<input checked="" type="checkbox"/>
triggerIndex	0>0	<input checked="" type="checkbox"/>
typeAnimated	<input type="checkbox"/>	<input checked="" type="checkbox"/>
typeBarrier	<input type="checkbox"/>	<input checked="" type="checkbox"/>
typeGate	<input type="checkbox"/>	<input checked="" type="checkbox"/>
typeLight	<input type="checkbox"/>	<input checked="" type="checkbox"/>



## Erklärung des Aufbaus im Scenegraph

- 0>**                    **gate**
- Objekt welches die User Attribute sowie den Scriptaufruf enthält

- 0>0**                    **gate\_trigger**
- Trigger (Kollisionsobjekt für Funktionsauslösung) welcher in den User Attributen angegeben wird (triggerIndex)

- 0>0|0**                    **holder**
- Objekt welches das zu bewegendes/rotierende Objekt enthält

- 0>0|0|0**                    **rollgate**
- Objekt welches bewegt werden soll (max# , min#)

- 0>1**                    **rollgate\_drive**
- Nicht weiter definiertes Objekt welches zur optischen Erweiterung des Gesamtobjektes beiträgt

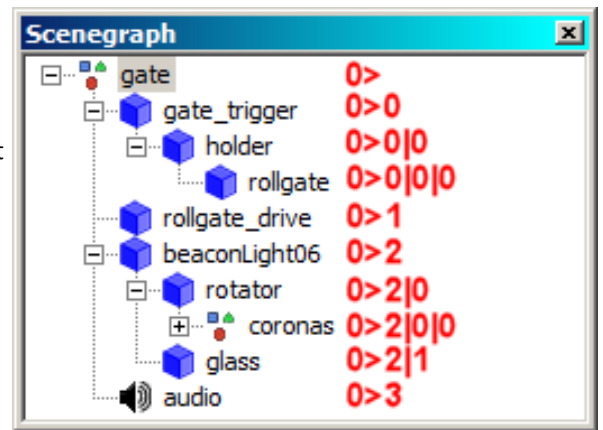
- 0>2**                    **beaconLight06**
- Objekt welches einem animierten Objekt als Träger dient

- 0>2|0**                    **rotator**
- Objekt welches eine Animation enthält (animationClip) und in den User Attributen angegeben wird (typeAnimated, animatorIndex)

- 0>2|0|0**                    **coronas**
- Objekt welches ein/ausblendbar dargestellt werden soll (typeLight) und in den User Attributen angegeben wird (lightIndex)

- 0>2|1**                    **glass**
- siehe 0>1

- 0>3**                    **audio**
- Objekt welches auf die AudioSource verweist die bei Bewegung abgespielt werden soll und in den User Attributen angegeben wird (audioIndex)



# Erklärung der MODDESC.XML Einträge

## 1.1

Folgende Zeilen müssen in die modDesc.xml aufgenommen werden:

```
<extraSourceFiles>
    <sourceFile filename="scripts/ManualBarrierManager.lua" />
    <sourceFile filename="scripts/ManualBarrier.lua" />
</extraSourceFiles>
```

Sollte bereits eine <extraSourceFiles>-Sektion in der modDesc.xml vorhanden sein sind lediglich die beiden <sourceFile>-Einträge zu übernehmen damit ManualBarrier funktioniert.

### 1.1.1

Wer die Scripte nicht in der modDesc.xml eintragen möchte, sie jedoch dennoch nutzbar machen will auf seiner Karte kann auch die entsprechende LUA der Karte (Beispiel: SampleModMap.lua) mit 2 Einträgen versehen:

```
source(Utils.getFilename("scripts/ManualBarrier.lua", baseDirectory));
source(Utils.getFilename("scripts/ManualBarrierManager.lua", baseDirectory));
```

Diese Einträge sind in der :new(baseDirectory, customMt) Funktion des Kartenscriptes einzutragen und zwar unmittelbar VOR dem return self der Funktion.

Beispiel:

```
function SampleModMap:new(baseDirectory, customMt)
    local mt = customMt;
    if mt == nil then
        mt = SampleModMap_mt;
    end;
    local self = SampleModMap:superClass():new(baseDirectory, mt);

    source(Utils.getFilename("scripts/ManualBarrier.lua", baseDirectory));
    source(Utils.getFilename("scripts/ManualBarrierManager.lua", baseDirectory));

    return self;
end;
```

## 1.2

Für das manuelle Bedienen ist eine Tastenkonfiguration in der modDesc.xml anzugeben:

```
<inputBindings>
    <input name="OPEN_GATE" category="VEHICLE ONFOOT" key1="KEY_o" />
</inputBindings>
```

Der Name "OPEN\_GATE" ist dabei unbedingt beizubehalten da dieser vom Script explizit abgefragt wird und es andernfalls zu Fehlermeldungen in der Log-Datei kommt.

Auch hier gilt: sollte es bereits eine <inputBindings>-Sektion in der modDesc.xml bestehen lediglich den <input>-Eintrag hinzufügen.



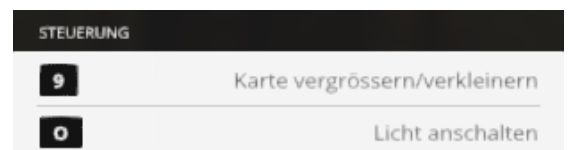
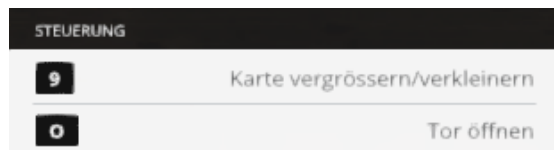
### 1.3

Standard Texteinträge für die mitgelieferten Beispielobjekte und Standard-Textzuweisungen des Scriptes sind in den L10N-Einträgen in der modDesc.xml hinzuzufügen:

<l10n>

```
<text name="string_OPEN"><en>open</en><de>öffnen</de></text>
<text name="string_CLOSE"><en>close</en><de>schließen</de></text>
<text name="string_ON"><en>switch on</en><de>anschalten</de></text>
<text name="string_OFF"><en>switch off</en><de>ausschalten</de></text>
<text name="string_BARRIER"><en>barrier</en><de>Schranke</de></text>
<text name="string_GATE"><en>gate</en><de>Tor</de></text>
<text name="string_LIGHT"><en>light</en><de>Licht</de></text>
<text name="string_DEFAULT"><en>gate/barrier</en><de>Tor/Schranke</de></text>
<text name="OPEN_GATE"><en>Open/close gate/door</en><de>Tor/Schranke öffnen/schließen</de></text>
```

</l10n>



Diese Werte sind die Standardwerte, eigene Einträge können selbstverständlich vorgenommen werden, wenn man zum Beispiel explizit ein Holztor angezeigt bekommen will kann man den Eintrag selbst hinzufügen:

```
<text name="string_WOODGATE"><en>wooden gate</en><de>Holztor</de></text>
```

Nachfolgend noch im I3D-Editor in den User Attributen den **string\_Name** Eintrag anpassen zu: string\_WOODGATE (Standard ist: string\_DEFAULT) und schon bekommt man im Spiel anstelle von „Tor/Schranke öffnen“ den Hilfeintrag „Holztor öffnen“ angezeigt.

**Auch hier gilt: sollte es bereits eine <l10n>-Sektion in der modDesc.xml bestehen lediglich die <text>-Eintrag hinzufügen.**

## Tipps

Wer ein animiertes Objekt zeitgleich mit einem rotierenden oder bewegenden Objekt laufen lassen will könnte die folgenden Formeln hilfreich finden.

Die Zeit um eine Animation nur in der Zeit ablaufen zu lassen während sich ein Objekt bewegt berechnet sich wie folgt:

$$\text{animationDuration} = (\text{max\#} - \text{min\#}) * \text{speedScale} * 15$$

Die Zeit um eine Animation nur in der Zeit ablaufen zu lassen während ein Objekt rotiert berechnet sich wie folgt:

$$\text{animationDuration} = (\text{max\#Angle} - \text{min\#Angle}) * \text{speedScale} / 2$$

Bei der Berechnung der Rotation gibt es leider zwischen der Öffnung und der Schließung manchmal einen Unterschied, in meinen Tests kamen Differenzen von teilweise 10 bis 40 Millisekunden zustande. Dies kann bei genau abgestimmten Abläufen hin und wieder zu Verzögerungen zwischen Rotationsbewegung und Animation führen.

Richtlinie die ich dabei herausgefunden habe: je geringer die **speedScale** ist desto geringer sind die Differenzen.

Viel Spaß wünscht euch Blacky\_BPG