

## UserAttributes im UPK

Die Module im UPK teilen sich eine Grundfunktionalität – hauptsächlich die Speicherung und Abrufung von Füllständen. Diese werden in den Standard-UserAttributes (siehe 1.1) festgelegt und können in jeden Modul verwendet werden.

Die Module, die als Trigger fungieren, können zusätzlich zu den Standard-UserAttributes und ihren eigenen die Trigger-UserAttributes (siehe 1.2) nutzen. Dort wird festgelegt, wann der Trigger die Funktion ausführen soll.

Alle als „default“ gekennzeichneten Werte der UserAttributes brauchen nicht gesetzt werden, sie sind es standardmäßig schon.

## Changelog

### V0.8.0

- **Multiplayer**-Support hinzugefügt

### V0.7.12

- Modultyp **waterfilltrigger** hinzugefügt (2.19)
- Modultyp **liquidmanurefilltrigger** hinzugefügt (2.10)
- neu: **createFillType** (waterfilltrigger, 2.19.1)
- neu: **pricePerLiter** (waterfilltrigger, 2.19.2)
- neu: **statName** (waterfilltrigger, 2.19.3)
- neu: **createFillType** (liquidmanurefilltrigger, 2.10.1)
- neu: **pricePerLiter** (liquidmanurefilltrigger, 2.10.2)
- neu: **statName** (liquidmanurefilltrigger, 2.10.3)
- Beispielmод UPK\_WaterTest, UPK\_LiquidManureTest und UPK\_MilkTest hinzugefügt

### V0.7.11

- Modultyp **balertrigger** hinzugefügt (2.3)
- Modultyp **sprayerfilltrigger** hinzugefügt (2.15)
- Modultyp **gasstationtrigger** hinzugefügt (2.9)
- neu: **fillType** (balertrigger, 2.3.1)
- neu: **fillLitersPerSecond** (balertrigger, 2.3.2)
- neu: **createFillType** (balertrigger, 2.3.3)
- neu: **pricePerLiter** (balertrigger, 2.3.4)
- neu: **statName** (balertrigger, 2.3.5)
- neu: **createFillType** (sprayerfilltrigger, 2.15.1)
- neu: **pricePerLiter** (sprayerfilltrigger, 2.15.2)
- neu: **statName** (sprayerfilltrigger, 2.15.3)
- neu: **createFillType** (gasstationtrigger, 2.9.1)
- neu: **pricePerLiter** (gasstationtrigger, 2.9.2)
- neu: **statName** (gasstationtrigger, 2.9.3)
- Fix für emptytrigger

- neu: **spezielle Fülltypen** "newVehiclesCost", "newAnimalsCost", "constructionCost", "vehicleRunningCost", "propertyMaintenance", "wagePayment", "harvestIncome", "missionIncome" und "loanInterest" hinzugefügt
- Beispielmод UPK\_FertilizerTest und UPK\_FuelTest hinzugefügt

#### V0.7.10

- Modultyp **animator** hinzugefügt (2.2)
- neu: **moveTo** (animator, 2.2.1)
- neu: **movementDuration** (animator, 2.2.2)
- neu: **movementSpeedupPeriod** (animator, 2.2.3)
- neu: **movementSlowdownPeriod** (animator, 2.2.4)
- neu: **rewindMovementOnDisable** (animator, 2.2.5)
- neu: **rotationsPerSecond** (animator, 2.2.6)
- neu: **rotateTo** (animator, 2.2.7)
- neu: **rotationDuration** (animator, 2.2.8)
- neu: **rotationSpeedupPeriod** (animator, 2.2.9)
- neu: **rotationSlowdownPeriod** (animator, 2.2.10)
- neu: **rewindRotationOnDisable** (animator, 2.2.11)
- neu: **animationClip** (animator, 2.2.12)
- neu: **animationSpeed** (animator, 2.2.13)
- neu: **animationLoop** (animator, 2.2.14)
- neu: **rewindAnimationOnDisable** (animator, 2.2.15)
- Beispielmод UPK\_AnimatorTest hinzugefügt

#### V0.7.9

- neu: **onCreate** (base, 2.1.3)
- geändert: alle UPK-Beispielmодs

#### V0.7.8

- Modultyp **switcher** hinzugefügt (2.16)
- Modultyp **selltarget** hinzugefügt (2.14)
- Modultyp **parktrigger** hinzugefügt (2.12)
- Fix für mover
- neu: **fillTypes** (switcher, 2.16.1)
- neu: **fillTypeChoice** (switcher, 2.16.2)
- neu: **switchFillTypes** (switcher, 2.16.3)
- neu: **switchFillLevels** (switcher, 2.16.4)
- neu: **mode** (switcher, 2.16.5)
- neu: **hidingPosition** (switcher, 2.16.6)
- geändert: startVisibilityAt (mover, 2.11.17)
- geändert: stopVisibilityAt (mover, 2.11.18)
- Beispielmод UPK\_ParkTriggerTest hinzugefügt

#### V0.7.7

- Modultyp **mover** hinzugefügt (2.11)

- neu: **fillTypes** (mover, 2.11.1)
- neu: **fillTypeChoice** (mover, 2.11.2)
- neu: **startMovingAt** (mover, 2.11.3)
- neu: **stopMovingAt** (mover, 2.11.4)
- neu: **lowPosition** (mover, 2.11.5)
- neu: **highPosition** (mover, 2.11.6)
- neu: **lowerPosition** (mover, 2.11.7)
- neu: **higherPosition** (mover, 2.11.8)
- neu: **movingType** (mover, 2.11.9)
- neu: **startTurningAt** (mover, 2.11.10)
- neu: **stopTurningAt** (mover, 2.11.11)
- neu: **lowRotation** (mover, 2.11.12)
- neu: **highRotation** (mover, 2.11.13)
- neu: **lowerRotation** (mover, 2.11.14)
- neu: **higherRotation** (mover, 2.11.15)
- neu: **turningType** (mover, 2.11.16)
- neu: **startVisibilityAt** (mover, 2.11.17)
- neu: **stopVisibilityAt** (mover, 2.11.18)
- Beispielmod UPK\_MoverTest hinzugefügt

#### V0.7.6

- Modultyp **dumptrigger** hinzugefügt (2.5)
- neu: **acceptedFillTypes** (dumptrigger, 2.5.1)
- neu: **revenuePerLiter** (dumptrigger, 2.5.2)
- neu: **revenuesPerLiter** (dumptrigger, 2.5.3)
- neu: **statName** (dumptrigger, 2.5.4)
- Fix für initialFillLevels (Standard-UserAttributes, 1.1.7)
- Beispielmod UPK\_DumpTriggerTest hinzugefügt

#### V0.7.5

- Fix für convertFillTypes (Standard-UserAttributes, 0)
- Beispielmod UPK\_TipTriggerTest2 hinzugefügt

#### V0.7.4

- neu: Fülltypen-Behandlung für money, void, sun, rain und temperature
- Beispielmod UPK\_ProcessorTest3 hinzugefügt

#### V0.7.3

- neu: **Speichern und Laden von Füllständen**
- neu: **convertFillTypes** (Standard-UserAttributes, 0)
- Modultyp **emptytrigger** hinzugefügt (2.6)
- neu: **emptyFillTypes** (emptytrigger, 2.6.1)
- neu: **emptyLitersPerSecond** (emptytrigger, 2.6.2)
- neu: **revenuePerLiter** (emptytrigger, 2.6.3)
- neu: **revenuesPerLiter** (emptytrigger, 2.6.4)
- neu: **statName** (emptytrigger, 2.6.5)
- neu: **revenuePerLiter** (tiptrigger, 2.17.4)

- neu: **revenuesPerLiter** (tiptrigger, 2.17.5)
- neu: **statName** (tiptrigger, 2.17.6)
- Beispielmод UPK\_EmptyTriggerTest hinzugefügt

## V0.7.2

- Modultyp **filltrigger** hinzugefügt (2.8)
- neu: **fillType** (filltrigger, 2.8.1)
- neu: **fillLitersPerSecond** (filltrigger, 2.8.2)
- neu: **createFillType** (filltrigger, 2.8.3)
- neu: **pricePerLiter** (filltrigger, 2.8.4)
- neu: **statName** (filltrigger, 2.8.5)
- Beispielmод UPK\_FillTriggerTest hinzugefügt

## V0.7.1

- Modultyp **tiptrigger** hinzugefügt (2.17)
- neu: **acceptedFillTypes** (tiptrigger, 2.17.1)
- neu: **showNotAcceptedWarning** (tiptrigger, 2.17.2)
- neu: **showCapacityReachedWarning** (tiptrigger, 2.17.3)
- Beispielmод UPK\_TipTriggerTest hinzugefügt

## V0.7.0

- neue Art Füllstände zu verwalten

(Versionsnummern übersprungen)

## V0.1.3

- Modultyp **unspecified** hinzugefügt (2.18)
- geändert: **enableChildrenIfProcessing** (processor, 2.13.19)
- neu: **addIfProcessing** (processor, 2.13.16)
- neu: **emptyFillTypesIfProcessing** (processor, 2.13.17)
- neu: **enableChildrenIfNotProcessing** (processor, 2.13.20)
- neu: **disableChildrenIfProcessing** (processor, 2.13.21)
- neu: **disableChildrenIfNotProcessing** (processor, 2.13.17)
- umbenannt: von „equal“ zu „uniform“ in **outcomeVariationType** (processor, 2.13.11)

# 1. Allgemeine UserAttributes

## 1.1. Standard-UserAttributes

### 1.1.1. type (string)

Gibt an, um welches Modul es sich handelt, d.h. welche Funktion es haben soll.

Ausprägung	Beschreibung
base	siehe 2.1
animator	siehe 2.2
balertrigger	siehe 2.3
displaytrigger	siehe 2.4
dumptrigger	siehe 2.5
emptytrigger	siehe 2.6
entitytrigger	siehe 2.7
filltrigger	siehe 2.8
gasstationtrigger	siehe 2.9
liquidmanurefilltrigger	siehe 2.10
mover	siehe 2.11
parktrigger	siehe 2.12
processor	siehe 2.13
selltarget	siehe 2.14
sprayerfilltrigger	siehe 2.15
switcher	siehe 2.16
tiptrigger	siehe 2.17
unspecified	siehe 2.18
waterfilltrigger	siehe 2.19

### 1.1.2. adjustToTerrainHeight (boolean)

Dieses UserAttribute ist auch für alle Nodes oder TriggerGroups im Mod anwendbar, die kein UPK-Modul sind.

Ausprägung	Beschreibung
true	Verschiebt den Node-Ursprung auf Höhe des Terrains.
false (default)	Verändert die Lage der Node nicht.

**1.1.3. convertFillTypes (string)**

Ausprägung	Beschreibung
(ohne) (default)	Keine Fülltypen werden umgewandelt
(Auflistung von Fülltypen)	<p>Auflistung von Fülltypen, die jeweils in den zuerst aufgeführten umgewandelt werden sollen. Mehrere Auflistungen können durch Komma getrennt werden.</p> <p>Am besten in dem Modul verwenden, wo der Ziel-Fülltyp gespeichert wird (siehe store 1.1.4).</p> <p>Hinweis: Die umzuwandelnden Fülltypen müssen trotzdem bei Triggern als akzeptiert aufgeführt werden.</p> <p>Bsp: Jede Strohart soll als Weizenstroh gespeichert werden: „wheat_windrow oat_windrow rice_windrow“</p> <p>Zudem soll Gras als Heu gespeichert werden: „wheat_windrow oat_windrow rice_windrow, dryGrass_windrow grass_windrow“</p>

**1.1.4. store (string)**

Regelt die Speicherung und Verwaltung von Füllständen.

Ausprägung	Beschreibung
(ohne) (default)	<p>Das Modul greift auf den selben Speicher zu wie das übergeordnete Modul und nutzt dessen Einstellungen.</p> <p>Nicht kombinierbar mit capacity oder capacities.</p> <p>Die Basis (siehe 2.1) speichert standardmäßig jeden Fülltyp separat ab.</p>
(Auflistung von Fülltypen)	<p>Die Speicherung erfolgt separat für jeden aufgelisteten Fülltyp. Bei nicht aufgelisteten Fülltypen wird an das übergeordnete Modul weiterverwiesen.</p> <p>Kombinierbar mit capacities (siehe 1.1.6)</p> <p>Bsp: „wheat“, „barley water“, „manure wheat barley“</p>
single	<p>Speichert nur einen einzigen Fülltyp ab. Sobald ein Füllstand gespeichert ist, ist auch dessen Fülltyp fixiert. Weitere Füllstände anderer Fülltypen können dann nicht mehr gespeichert werden. Der Fülltyp kann sich erst ändern, wenn der Füllstand des vorherigen wieder leer ist.</p> <p>Kombinierbar mit capacities (siehe 1.1.6)</p>
fifo	<p>„first in – first out“: Füllstände, die zuerst in dem Modul gespeichert werden, werden als erste wieder ausgegeben.</p> <p>Funktion ähnlich zu einem Rohr: Wenn man Weizen, Gerste und Raps in ein Rohr kippt, kommt auf der anderen Seite Weizen, Gerste und Raps in dieser Reihenfolge wieder raus.</p>
filo	<p>„first in – last out“: Füllstände, die zuerst in dem Modul gespeichert werden, werden als letzte wieder ausgegeben.</p> <p>Funktion ähnlich zu einem Fass: Wenn man Weizen, Gerste und Raps in ein Fass kippt, muss man Raps, Gerste und Weizen in dieser Reihenfolge wieder entnehmen, um es zu leeren.</p>

### 1.1.5. capacity (float)

Legt den maximalen Standard-Füllstand fest.

Bei separater und einzelner Füllstandsspeicherung gilt dies für jeden Fülltyp, für fifo und filo nur insgesamt.

Ausprägung	Beschreibung
(ohne) (default)	Die Füllmenge ist unendlich.
(Zahl)	Maximaler Füllstands-Wert Bsp: „30000“, „6500“

### 1.1.6. capacities (string)

Ausprägung	Beschreibung
(ohne) (default)	Keine Ausnahmen für capacity (siehe 1.1.5)
(Auflistung von Füllständen und Fülltypen)	Legt die Füllstände für verschiedene Fülltypen fest. Für alle hier nicht aufgelisteten Fülltypen, die gespeichert werden, gilt capacity (siehe 1.1.5). Nicht in Kombination mit store=“fifo“ oder store=“filo“. Bsp: „6000 wheat“, „5000 water 1000 manure“

### 1.1.7. initialFillLevels (string)

Ausprägung	Beschreibung
(ohne) (default)	Alle Füllmengen sind beim Kauf des Mods/ zu Beginn des Spiels 0.
(Auflistung von Füllständen und Fülltypen)	Setzt bestimmte Fülltypen einmalig beim Kauf/ beim Beginn des Spiels auf die angegebenen Werte. Bsp: „500 wheat“, „300 water 60 manure“

### 1.1.8. isEnabled (boolean)

Legt fest, ob das Modul beim Kauf des Mods/ zu Beginn des Spiels aktiviert oder deaktiviert ist.

Ausprägung	Beschreibung
true (default)	das Modul ist aktiviert
false	das Modul ist deaktiviert

## 1.2. Trigger-UserAttributes

### 1.2.1. allowWalker (boolean)

Ausprägung	Beschreibung
true (default)	der Trigger funktioniert für Fußgänger
false	der Trigger funktioniert nicht für Fußgänger

### 1.2.2. allowMotorized (boolean)

Ausprägung	Beschreibung
true (default)	der Trigger funktioniert für Fahrzeuge mit Motor
false	der Trigger funktioniert nicht für Fahrzeuge mit Motor

### 1.2.3. allowCombine (boolean)

Ausprägung	Beschreibung
true (default)	der Trigger funktioniert für Erntemaschinen
false	der Trigger funktioniert nicht für Erntemaschinen

### 1.2.4. allowFillable (boolean)

Als Überkategorie für alle speziellen Anhänger.

Ausprägung	Beschreibung
true (default)	der Trigger funktioniert für alle Anhänger
false	der Trigger funktioniert nur für diejenigen Anhänger, die mit dem entsprechenden UserAttribute auf true gesetzt sind

Bsp um nur Ballenwickler und Ladewagen zu akzeptieren:

allowFillable=false  
allowBaler=true  
allowForageWagon=true

### 1.2.5. allowTipper (boolean)

Ausprägung	Beschreibung
true	der Trigger funktioniert für Kipper
false (default)	der Trigger funktioniert nicht für Kipper

### 1.2.6. allowShovel (boolean)

Ausprägung	Beschreibung
true	der Trigger funktioniert für Schaufeln
false (default)	der Trigger funktioniert nicht für Schaufeln



**1.2.7. allowWaterTrailer (boolean)**

Ausprägung	Beschreibung
true	der Trigger funktioniert für Wassertanks
false (default)	der Trigger funktioniert nicht für Wassertanks

**1.2.8. allowFuelTrailer (boolean)**

Ausprägung	Beschreibung
true	der Trigger funktioniert für Treibstofftanks
false (default)	der Trigger funktioniert nicht für Treibstofftanks

**1.2.9. allowLiquidManureTrailer (boolean)**

Ausprägung	Beschreibung
true	der Trigger funktioniert für Gülletanks
false (default)	der Trigger funktioniert nicht für Gülletanks

**1.2.10. allowMilkTrailer (boolean)**

Ausprägung	Beschreibung
true	der Trigger funktioniert für Milchtanks
false (default)	der Trigger funktioniert nicht für Milchtanks

**1.2.11. allowSowingMachine (boolean)**

Ausprägung	Beschreibung
true	der Trigger funktioniert für Sämaschinen
false (default)	der Trigger funktioniert nicht für Sämaschinen

**1.2.12. allowSprayer (boolean)**

Ausprägung	Beschreibung
true	der Trigger funktioniert für Spritzen
false (default)	der Trigger funktioniert nicht für Spritzen

**1.2.13. allowForageWagon (boolean)**

Ausprägung	Beschreibung
true	der Trigger funktioniert für Ladewagen
false (default)	der Trigger funktioniert nicht für Ladewagen

**1.2.14. allowBaler (boolean)**

<b>Ausprägung</b>	<b>Beschreibung</b>
true	der Trigger funktioniert für Ballenpressen/-wickler
false (default)	der Trigger funktioniert nicht für Ballenpressen/-wickler

**1.2.15. allowTrafficVehicle (boolean)**

<b>Ausprägung</b>	<b>Beschreibung</b>
true	der Trigger funktioniert für Wagen des automatischen Straßenverkehrs
false (default)	der Trigger funktioniert nicht für Wagen des automatischen Straßenverkehrs

## 2. spezielle UserAttributes

### 2.1. base

Dieser Modultyp wird für das oberste Modul im Mod automatisch vergeben und kann nicht manuell verwendet werden. D.h. alle unten aufgeführten UserAttributes gelten nur für das oberste Modul im Mod.

#### 2.1.1. UPKversion (string)

Ausprägung	Beschreibung
(ohne) (default)	keine Überprüfung der Versionsnummer
(Zahl.Zahl.Zahl)	Mindest zu verwendende UPK-Version. Falls diese vom Nutzer unterschritten wird, wird ein Fehler angezeigt und der Mod nicht geladen. Bsp: „0.1.1“

#### 2.1.2. modname (string)

Ausprägung	Beschreibung
(ohne) (default)	
(Name des Mods)	Gibt den Namen des Mods bzw. des verwendeten Mod-Ordners an. Für verschiedene Dinge wichtig, z.Bsp. für Extraeintragungen in der ModDesc.xml. Groß- und Kleinschreibung beachten.

#### 2.1.3. onCreate (scriptCallback)

Ausprägung	Beschreibung
(ohne) (default)	
„UniversalProcessKit.onCreate“	Lädt in Karten fest verbaute UPK-Mods. Es hat keine Auswirkung bei platzierbaren UPK-Mods.

## 2.2. animator

Der animator bewegt Shapes. Anders als der mover, der auf Füllstände reagiert, spielt der animator die Animation ab, sobald er aktiviert ist. Es lassen sich 4 Möglichkeiten einstellen, die einzeln oder zusammen benutzt werden können: Eine Bewegung zu einer bestimmten Position, eine ständige Drehung, eine Drehung zu einer bestimmten Ausrichtung oder das Abspielen einer Animation.

In Kombination mit dem entitytrigger ergeben sich Möglichkeiten wie Tore, Schranken oder Fahrstühle. Auch der processor kann den animator aktivieren.

### 2.2.1. moveTo (string)

Ausprägung	Beschreibung
„0 0 0“ (default)	Das Shape wird nicht verschoben.
„Zahl Zahl Zahl“	Gibt die Position des Shapes an, die beim Ende der Bewegung erreicht sein soll (X Y Z).

### 2.2.2. movementDuration (float)

Ausprägung	Beschreibung
0 (default)	
Zahl	Gibt die Dauer der Bewegung in Sekunden an.

### 2.2.3. movementSpeedupPeriod (float)

Ausprägung	Beschreibung
0 (default)	
Zahl zw. 0 und 1	Gibt den Anteil der Bewegungsdauer an, bei dem die Bewegung am Anfang beschleunigt werden soll. Bspw. „0.2“ entspricht 20% der Dauer soll zuerst beschleunigt werden. Die Bewegung sieht mit diesem Attribut „natürlicher“ aus. Die Summe von movementSpeedupPeriod und movementSlowdownPeriod ist maximal 1.

### 2.2.4. movementSlowdownPeriod (float)

Ausprägung	Beschreibung
0 (default)	
Zahl zw. 0 und 1	Gibt den Anteil der Bewegungsdauer an, bei dem die Bewegung am Ende abgebremst werden soll. Bspw. „0.2“ entspricht 20% der Dauer soll am Ende abgebremst werden. Die Bewegung sieht mit diesem Attribut „natürlicher“ aus. Die Summe von movementSpeedupPeriod und movementSlowdownPeriod ist maximal 1.

### 2.2.5. rewindMovementOnDisable (boolean)

Ausprägung	Beschreibung
true (default)	Die Bewegung läuft rückwärts, wenn das Modul deaktiviert wird. Bspw. in Verwendung mit einem entitytrigger und einem Tor, das sich beim Verlassen des Triggers wieder schließt.
false	Die Bewegung bleibt auf ihrem Stand beim Verlassen.

### 2.2.6. rotationsPerSecond (string)

Ausprägung	Beschreibung
„0 0 0“ (default)	Das Shape dreht sich nicht.
„Zahl Zahl Zahl“	Gibt an, wie schnell sich das Shape pro Sekunde drehen soll. Steht nicht in Zusammenhang mit rotateTo und dessen weitere UserAttributes.

### 2.2.7. rotateTo (string)

Ausprägung	Beschreibung
„0 0 0“ (default)	Das Shape dreht sich nicht.
„Zahl Zahl Zahl“	Gibt an, wie das Shape am Ende der Drehung ausgerichtet sein soll. Steht nicht in Zusammenhang mit rotationsPerSecond.

### 2.2.8. rotationDuration (float)

Ausprägung	Beschreibung
0 (default)	
Zahl	Gibt die Dauer der Drehung in Sekunden an.

### 2.2.9. rotationSpeedupPeriod (float)

Ausprägung	Beschreibung
0 (default)	
Zahl zw. 0 und 1	Gibt den Anteil der Rotationsdauer an, bei dem die Drehung am Anfang beschleunigt werden soll. Bspw. „0.2“ entspricht 20% der Dauer soll zuerst beschleunigt werden. Die Drehung sieht mit diesem Attribut „natürlicher“ aus. Die Summe von rotationSpeedupPeriod und rotationSlowdownPeriod ist maximal 1.

### 2.2.10. rotationSlowdownPeriod (float)

Ausprägung	Beschreibung
0 (default)	
Zahl zw. 0 und 1	Gibt den Anteil der Rotationsdauer an, bei dem die Drehung am Ende abgebremst werden soll. Bspw. „0.2“ entspricht 20% der Dauer soll am Ende abgebremst werden. Die Drehung sieht mit diesem Attribut „natürlicher“ aus. Die Summe von rotationSpeedupPeriod und rotationSlowdownPeriod ist maximal 1.

### 2.2.11. rewindRotationOnDisable (boolean)

Ausprägung	Beschreibung
true (default)	Die Drehung läuft rückwärts, wenn das Modul deaktiviert wird. Bspw. in Verwendung mit einem entitytrigger und einer Schranke, die sich beim Verlassen des Triggers wieder schließt.
false	Die Bewegung bleibt auf ihrem Stand beim Verlassen.

### 2.2.12. animationClip (string)

Ausprägung	Beschreibung
(ohne) (default)	Kein Clip wird abgespielt.
„Name des Clips“	Name des Clips, der abgespielt werden soll.

### 2.2.13. animationSpeed (float)

Ausprägung	Beschreibung
1 (default)	Der Clip wird mit der normalen Geschwindigkeit abgespielt.
Zahl	Der Clip wird langsamer (<1) oder schneller (>1) abgespielt.

### 2.2.14. animationLoop (boolean)

Ausprägung	Beschreibung
true	Der Clip wird in einer Schleife dauernd abgespielt.
false (default)	Der Clip wird nicht in einer Schleife abgespielt.

### 2.2.15. rewindAnimationOnDisable (boolean)

Ausprägung	Beschreibung
true	Der Clip läuft rückwärts, wenn das Modul deaktiviert ist.
false (default)	Der Clip bleibt in seiner Position, wenn das Modul deaktiviert ist.

## 2.3. balertrigger

Dieser Trigger füllt Ballenwickler und Ladewagen über den PickUp-Mechanismus wenn er angeschaltet ist

Die Default-Werte der Trigger-UserAttributes sind folgende:

UserAttribute	Default-Wert
allowBaler	true
allowForageWagon	true
(andere)	false

### 2.3.1. fillType (string)

Ausprägung	Beschreibung
(ohne) (default)	Nichts wird abgefüllt.
(Fülltyp)	Der Fülltyp, der abgefüllt werden soll. Bsp: „grass_windrow“, „wheat_windrow“

### 2.3.2. fillLitersPerSecond (float)

Ausprägung	Beschreibung
1500 (default)	Die Befüllgeschwindigkeit ist 1500 Liter pro Sekunde.
(Zahl)	Legt die Befüllgeschwindigkeit in Liter pro Sekunde fest.

### 2.3.3. createFillType (boolean)

Ausprägung	Beschreibung
true	Legt fest, dass die Befüllung unabhängig vom Füllstand stattfindet.
false (default)	Legt fest, dass die abgefüllte Füllmenge vorhanden sein muss.

### 2.3.4. pricePerLiter (float)

Ausprägung	Beschreibung
0 (default)	Die Befüllung ist kostenlos.
(Zahl)	Die Befüllung kostet X € pro Liter. Negative Werte werden zum Kontostand hinzugefügt. Mit statName kombinierbar.

### 2.3.5. statName (string)

Falls die Abfüllung Geld kostet (oder einbringt), legt statName fest, auf welchen Punkt in der Spiel-Statistik der Betrag gebucht werden soll.

<b>Ausprägung</b>	<b>Beschreibung</b>
„other“ (default)	Sonstiges
„newVehiclesCost“	Neue Fahrzeuge
„newAnimalsCost“	Tierkosten
„constructionCost“	Baukosten
„vehicleRunningCost“	Betriebskosten Fahrzeuge
„propertyMaintenance“	Betriebskosten Gebäude
„wagePayment“	Lohnzahlungen
„harvestIncome“	Einnahmen Ernte
„missionIncome“	Einnahmen Missionen
„loanInterest“	Kreditzinsen



## 2.4. displaytrigger

Für dieses Modul gelten auch die Trigger-UserAttributes (siehe 1.2).

Die Default-Werte der Trigger-UserAttributes sind folgende:

UserAttribute	Default-Wert
allowWalker	true
allowMotorized	true
allowFillable	true
allowCombine	true
(andere)	false

### 2.4.1. displayFillTypes (string)

Ausprägung	Beschreibung
(ohne) (default)	Keine Füllstände werden angezeigt
(Auflistung von Füllständen)	Eine Liste mit anzuzeigenden Füllsorten, durch Leerzeichen getrennt. Bsp: „wheat“, „wheat barley“, „wheat barley water manure“

### 2.4.2. onlyFilled (boolean)

Ausprägung	Beschreibung
true (default)	zeigt nur die angegebenen Füllsorten (displayFillTypes) an, wenn deren Füllstand größer als 0 Liter ist
false	zeigt immer jeden Füllstand an, auch 0 Liter

### 2.4.3. showFillLevel (boolean)

Ausprägung	Beschreibung
true (default)	zeigt den Füllstand in Litern an
false	zeigt ihn nicht an

### 2.4.4. showPercentage (boolean)

Ausprägung	Beschreibung
true (default)	zeigt den Füllstand in Relation zur Kapazität (capacity) in Prozent an
false	zeigt ihn nicht an

## 2.5. dumptrigger

Dieser „Trigger“ ist ein flaches Objekt am Boden und fängt die Entladung von Schaufeln und Erntemaschinen auf.

Es ist kein Trigger im eigentlichen Sinn (da nichts in ihn reinfahren muss), weswegen nur die Standard-UserAttributes hinzukommen, nicht die für normale Trigger.

### 2.5.1. acceptedFillTypes (string)

Ausprägung	Beschreibung
(ohne) (default)	Nichts wird akzeptiert.
(Auflistung von Füllständen)	Eine Liste mit zu akzeptierenden Fülltypen. Bsp: „wheat“, „wheat barley“, „wheat barley rape maize“

### 2.5.2. revenuePerLiter (float)

revenuePerLiter gilt für alle erlaubten Fruchttypen. Davon kann man mit revenuesPerLiter Ausnahmen festlegen bzw. gleich alle mit Preisen auflisten. Zuerst schaut das Skript ob in revenuesPerLiter ein Wert für den Fruchttyp festgelegt wurde, sonst nimmt es den Wert von revenuePerLiter (per default auf 0€ pro Liter gesetzt).

ZBsp.: acceptedFillTypes="wheat barley maize rape", revenuePerLiter="0.1", revenuesPerLiter="1.2 wheat 1.3 barley" und convertFillTypes="void wheat rape".  
Für Weizen bekommt man 1,2€ pro Liter und es wird gelöscht. Für Barley bekommt man 1,3€ pro Liter und es wird gespeichert. Für Mais bekommt man 0,1€ und es wird gespeichert. Für Raps bekommt man auch 0,1€ und es wird gelöscht.

Ausprägung	Beschreibung
0 (default)	Das Abladen ist bringt kein Geld ein.
(Zahl)	Das Abladen bringt X € pro Liter ein. Negative Werte werden zum Kontostand hinzugefügt. Mit statName kombinierbar.

### 2.5.3. revenuesPerLiter (string)

Ausprägung	Beschreibung
(ohne) (default)	Das Abladen ist bringt kein Geld ein bzw. es gilt pricePerLiter.
(Auflistung von Kosten und Fülltypen)	Das Abladen bringt X € pro Liter ein. Negative Werte werden dem Kontostand hinzugefügt. Mit statName kombinierbar. Für nicht hier aufgeführte, aber im Trigger akzeptierte Fülltypen, gilt pricePerLiter. Bsp: „1.2 wheat 1.3 barley“

#### 2.5.4. statName (string)

Falls das Abladen Geld einbringt (oder kostet), legt statName fest, auf welchen Punkt in der Spiel-Statistik der Betrag gebucht werden soll.

<b>Ausprägung</b>	<b>Beschreibung</b>
„other“ (default)	Sonstiges
„newVehiclesCost“	Neue Fahrzeuge
„newAnimalsCost“	Tierkosten
„constructionCost“	Baukosten
„vehicleRunningCost“	Betriebskosten Fahrzeuge
„propertyMaintenance“	Betriebskosten Gebäude
„wagePayment“	Lohnzahlungen
„harvestIncome“	Einnahmen Ernte
„missionIncome“	Einnahmen Missionen
„loanInterest“	Kreditzinsen

## 2.6. emptytrigger

Dieser Trigger entleert Fahrzeuge und Anhänger ohne Animationen oder Geräusche derselben auszulösen wie die Abkippanimation (siehe tiptrigger) oder Tankgeräusche (siehe gasstationtrigger). Dieser Trigger ist quasi das Gegenteil des filltriggers.

Die Default-Werte der Trigger-UserAttributes sind folgende:

UserAttribute	Default-Wert
allowTipper	true
allowShovel	true
allowSowingMachine	true, wenn filltype="seeds"
allowWaterTrailer	true, wenn filltype="water"
allowMilkTrailer	true, wenn filltype="milk"
allowLiquidManureTrailer	true, wenn filltype="liquidManure"
allowSprayer	true, wenn filltype="fertilizer"
allowFuelTrailer	true, wenn filltype="fuel"
allowMotorized	false
(andere)	false

### 2.6.1. emptyFillTypes (string)

Ausprägung	Beschreibung
(ohne) (default)	Nichts wird abgefüllt.
(Auflistung von Fülltypen)	Auflistung von Fülltypen, die geleert werden sollen. Bsp: „wheat“, „barley maize“, „water fuel wheat“

### 2.6.2. emptyLitersPerSecond (float)

Ausprägung	Beschreibung
1500 (default)	Die Entleerungsgeschwindigkeit ist 1500 Liter pro Sekunde.
(Zahl)	Legt die Entleerungsgeschwindigkeit in Liter pro Sekunde fest.

### 2.6.3. revenuePerLiter (float)

revenuePerLiter gilt für alle erlaubten Fruchttypen. Davon kann man mit revenuesPerLiter Ausnahmen festlegen bzw. gleich alle mit Preisen auflisten. Zuerst schaut das Skript ob in revenuesPerLiter ein Wert für den Fruchttyp festgelegt wurde, sonst nimmt es den Wert von revenuePerLiter (per default auf 0€ pro Liter gesetzt).

ZBsp.: acceptedFillTypes="wheat barley maize rape", revenuePerLiter="0.1", revenuesPerLiter="1.2 wheat 1.3 barley" und convertFillTypes="void wheat rape".  
Für Weizen bekommt man 1,2€ pro Liter und es wird gelöscht. Für Barley bekommt man 1,3€ pro Liter und es wird gespeichert. Für Mais bekommt man 0,1€ und es wird gespeichert. Für Raps bekommt man auch 0,1€ und es wird gelöscht.

Ausprägung	Beschreibung
0 (default)	Die Entleerung ist ohne Gewinn.
(Zahl)	Die Entleerung bringt X € pro Liter ein. Negative Werte werden zum Kontostand hinzugefügt. Mit statName kombinierbar.

#### 2.6.4. revenuesPerLiter (string)

Ausprägung	Beschreibung
(ohne) (default)	Die Entleerung ist kostenlos bzw. es gilt pricePerLiter.
(Auflistung von Kosten und Fülltypen)	Die Entleerung kostet X € pro Liter. Negative Werte werden dem Kontostand hinzugefügt. Mit statName kombinierbar. Für nicht hier aufgeführte, aber im Trigger akzeptierte Fülltypen, gilt pricePerLiter. Bsp: „1.2 wheat 1.3 barley“

#### 2.6.5. statName (string)

Falls die Abfüllung Geld kostet (oder einbringt), legt statName fest, auf welchen Punkt in der Spiel-Statistik der Betrag gebucht werden soll.

Ausprägung	Beschreibung
„other“ (default)	Sonstiges
„newVehiclesCost“	Neue Fahrzeuge
„newAnimalsCost“	Tierkosten
„constructionCost“	Baukosten
„vehicleRunningCost“	Betriebskosten Fahrzeuge
„propertyMaintenance“	Betriebskosten Gebäude
„wagePayment“	Lohnzahlungen
„harvestIncome“	Einnahmen Ernte
„missionIncome“	Einnahmen Missionen
„loanInterest“	Kreditzinsen

## 2.7. entitytrigger

Aktiviert und deaktiviert die untergeordneten Module.

Zum Beispiel um eine Animation, wie eine Schranke, abzuspielen, wenn sich ein Fahrzeug davor befindet.

Die Default-Werte der Trigger-UserAttributes sind folgende:

UserAttribute	Default-Wert
allowWalker	true
allowMotorized	true
allowFillable	true
allowCombine	true
(andere)	false

### 2.7.1. enableOnEmpty (boolean)

Ausprägung	Beschreibung
true	aktiviert die untergeordneten Module wenn sich nichts im Trigger befindet
false (default)	aktiviert die untergeordneten Module wenn sich etwas im Trigger befindet

## 2.8. filltrigger

Dieser Trigger befüllt Fahrzeuge und Anhänger ohne Animationen oder Geräusche derselben auszulösen wie die Tankgeräusche (siehe gasstationtrigger). Dieser Trigger ist quasi das Gegenteil des emptytriggers.

Die Default-Werte der Trigger-UserAttributes sind folgende:

UserAttribute	Default-Wert
allowTipper	true
allowShovel	true
allowSowingMachine	true, wenn filltype="seeds"
allowWaterTrailer	true, wenn filltype="water"
allowMilkTrailer	true, wenn filltype="milk"
allowLiquidManureTrailer	true, wenn filltype="liquidManure"
allowSprayer	true, wenn filltype="fertilizer"
allowFuelTrailer	true, wenn filltype="fuel"
allowMotorized	false
(andere)	false

### 2.8.1. fillType (string)

Ausprägung	Beschreibung
(ohne) (default)	Nichts wird abgefüllt.
(Fülltyp)	Der Fülltyp, der abgefüllt werden soll. Bsp: „wheat“, „barley“, „water“

### 2.8.2. fillLitersPerSecond (float)

Ausprägung	Beschreibung
1500 (default)	Die Befüllgeschwindigkeit ist 1500 Liter pro Sekunde.
(Zahl)	Legt die Befüllgeschwindigkeit in Liter pro Sekunde fest.

### 2.8.3. createFillType (boolean)

Ausprägung	Beschreibung
true	Legt fest, dass die Befüllung unabhängig vom Füllstand stattfindet.
false (default)	Legt fest, dass die abgefüllte Füllmenge vorhanden sein muss.

### 2.8.4. pricePerLiter (float)

Ausprägung	Beschreibung
0 (default)	Die Befüllung ist kostenlos.
(Zahl)	Die Befüllung kostet X € pro Liter. Negative Werte werden zum Kontostand hinzugefügt. Mit statName kombinierbar.

### 2.8.5. statName (string)

Falls die Abfüllung Geld kostet (oder einbringt), legt statName fest, auf welchen Punkt in der Spiel-Statistik der Betrag gebucht werden soll.

Ausprägung	Beschreibung
„other“ (default)	Sonstiges
„newVehiclesCost“	Neue Fahrzeuge
„newAnimalsCost“	Tierkosten
„constructionCost“	Baukosten
„vehicleRunningCost“	Betriebskosten Fahrzeuge
„propertyMaintenance“	Betriebskosten Gebäude
„wagePayment“	Lohnzahlungen
„harvestIncome“	Einnahmen Ernte
„missionIncome“	Einnahmen Missionen
„loanInterest“	Kreditzinsen



## 2.9. gasstationtrigger

Dieser Trigger befüllt Fahrzeuge und Tankanhänger mit Treibstoff. Im Gegensatz zum filltrigger muss die Abfüllung vom Fahrzeug aus bestätigt werden, genauso wie die Standard-Trigger.

Die Default-Werte der Trigger-UserAttributes sind folgende:

UserAttribute	Default-Wert
allowMotorized	true
allowFuelTrailer	true
(andere)	false

### 2.9.1. createFillType (boolean)

Ausprägung	Beschreibung
true	Legt fest, dass die Befüllung unabhängig vom Füllstand stattfindet.
false (default)	Legt fest, dass die abgefüllte Füllmenge vorhanden sein muss.

### 2.9.2. pricePerLiter (float)

Ausprägung	Beschreibung
0 (default)	Die Befüllung ist kostenlos.
(Zahl)	Die Befüllung kostet X € pro Liter. Negative Werte werden zum Kontostand hinzugefügt. Mit statName kombinierbar.

### 2.9.3. statName (string)

Falls die Abfüllung Geld kostet (oder einbringt), legt statName fest, auf welchen Punkt in der Spiel-Statistik der Betrag gebucht werden soll.

Ausprägung	Beschreibung
„other“ (default)	Sonstiges
„newVehiclesCost“	Neue Fahrzeuge
„newAnimalsCost“	Tierkosten
„constructionCost“	Baukosten
„vehicleRunningCost“	Betriebskosten Fahrzeuge
„propertyMaintenance“	Betriebskosten Gebäude
„wagePayment“	Lohnzahlungen
„harvestIncome“	Einnahmen Ernte
„missionIncome“	Einnahmen Missionen
„loanInterest“	Kreditzinsen

## 2.10. liquidmanurefilltrigger

Dieser Trigger befüllt Güllefässer mit Gülle. Im Gegensatz zum filltrigger muss die Abfüllung vom Fahrzeug aus bestätigt werden, genauso wie die Standard-Trigger.

Die Default-Werte der Trigger-UserAttributes sind folgende:

UserAttribute	Default-Wert
allowLiquidManureTrailer	true
(andere)	false

### 2.10.1. createFillType (boolean)

Ausprägung	Beschreibung
true	Legt fest, dass die Befüllung unabhängig vom Füllstand stattfindet.
false (default)	Legt fest, dass die abgefüllte Füllmenge vorhanden sein muss.

### 2.10.2. pricePerLiter (float)

Ausprägung	Beschreibung
0 (default)	Die Befüllung ist kostenlos.
(Zahl)	Die Befüllung kostet X € pro Liter. Negative Werte werden zum Kontostand hinzugefügt. Mit statName kombinierbar.

### 2.10.3. statName (string)

Falls die Abfüllung Geld kostet (oder einbringt), legt statName fest, auf welchen Punkt in der Spiel-Statistik der Betrag gebucht werden soll.

Ausprägung	Beschreibung
„other“ (default)	Sonstiges
„newVehiclesCost“	Neue Fahrzeuge
„newAnimalsCost“	Tierkosten
„constructionCost“	Baukosten
„vehicleRunningCost“	Betriebskosten Fahrzeuge
„propertyMaintenance“	Betriebskosten Gebäude
„wagePayment“	Lohnzahlungen
„harvestIncome“	Einnahmen Ernte
„missionIncome“	Einnahmen Missionen
„loanInterest“	Kreditzinsen

## 2.11. mover

Der mover verschiebt oder dreht Shapes und TransformGroups je nach Füllstand der angegebenen Fruchttypen. Er kann sie auch ein- oder ausblenden.

### 2.11.1. fillTypes (string)

Ausprägung	Beschreibung
<i>(ohne)</i> (default)	Auf nichts wird reagiert.
<i>(Auflistung von Fülltypen)</i>	Die Fülltypen, auf die reagiert werden soll. Bsp: „wheat“, „wheat barley“

### 2.11.2. fillTypeChoice (string)

Ausprägung	Beschreibung
„max“ (default)	Wählt den Füllstand desjenigen Fülltyps aus, der am höchsten ist.
„min“	Wählt den Füllstand desjenigen Fülltyps aus, der am niedrigsten ist.

### 2.11.3. startMovingAt (float)

Ausprägung	Beschreibung
0 (default)	Das Verschieben startet bei 0 Litern Füllmenge.
<i>(Zahl)</i>	Das Verschieben startet bei X Litern Füllmenge.

### 2.11.4. stopMovingAt (float)

Ausprägung	Beschreibung
größte Kapazität der Fülltypen (default)	Das Verschieben endet bei der höchsten Maximalfüllmenge der Fülltypen.
<i>(Zahl)</i>	Das Verschieben endet bei X Litern Füllmenge.

### 2.11.5. lowPosition (string)

Ausprägung	Beschreibung
„0 0 0“ (default)	Das Verschieben startet beim Ursprung des Shapes
„Zahl Zahl Zahl“	Gibt die Position des Shapes an, von der das Verschieben starten soll (X Y Z).

### 2.11.6. highPosition (string)

Ausprägung	Beschreibung
„0 0 0“ (default)	Das Verschieben endet beim Ursprung des Shapes
„Zahl Zahl Zahl“	Gibt die Position des Shapes an, bei der das Verschieben enden soll (X Y Z).

### 2.11.7. lowerPosition (string)

Ausprägung	Beschreibung
„0 0 0“ (default)	Das Shape bleibt bei seinem Ursprung, wenn der Füllstand kleiner ist als startMovingAt (X Y Z).
„Zahl Zahl Zahl“	Gibt die Position des Shapes an, wenn der Füllstand kleiner ist als startMovingAt (X Y Z). Bspw. um eine Plane zu verbergen/ unterhalb der Karte zu schieben, wenn der Füllstand kleiner als 1 Liter ist.

### 2.11.8. higherPosition (string)

Ausprägung	Beschreibung
highPosition (default)	Das Shape verbleibt bei highPosition, wenn der Füllstand größer ist als stopMovingAt (X Y Z).
„Zahl Zahl Zahl“	Gibt die Position des Shapes an, wenn der Füllstand größer ist als stopMovingAt (X Y Z).

### 2.11.9. movingType (string)

Art die Strecke zwischen lowPosition und highPosition in Relation zur Füllstandsänderung zurückzulegen.

Ausprägung	Beschreibung
„linear“ (default)	linear: Bspw für eine Füllplane in einer Box oder einem Fass
„sphere“	wie eine Halbkugel: Das dargestellte Volumen entspricht in etwa einer Halbkugel, wie bspw. ein Haufen
„cone“	wie ein Kegel: Das dargestellte Volumen entspricht in etwa einem Kegel, wie bspw. ein Rübenhaufen
„square“	quadratisch
„circle“	wie ein Kreis
„sinus“	anhand einer Sinus-Funktion

### 2.11.10. startTurningAt (float)

Ausprägung	Beschreibung
0 (default)	Das Drehen startet bei 0 Litern Füllmenge.
(Zahl)	Das Drehen startet bei X Litern Füllmenge.

### 2.11.11. stopTurningAt (float)

Ausprägung	Beschreibung
größte Kapazität der Fülltypen (default)	Das Drehen endet bei der höchsten Maximalfüllmenge der Fülltypen.
(Zahl)	Das Drehen endet bei X Litern Füllmenge.

### 2.11.12. lowRotation (string)

Ausprägung	Beschreibung
„0 0 0“ (default)	Das Drehen startet bei der ursprünglichen Drehung des Shapes
„Zahl Zahl Zahl“	Gibt die Drehung des Shapes an, von der das Drehen starten soll (X Y Z).

### 2.11.13. highRotation (string)

Ausprägung	Beschreibung
„0 0 0“ (default)	Das Drehen endet bei der ursprünglichen Drehung des Shapes
„Zahl Zahl Zahl“	Gibt die Drehung des Shapes an, bei der das Drehen enden soll (X Y Z).

### 2.11.14. lowerRotation (string)

Ausprägung	Beschreibung
„0 0 0“ (default)	Das Shape bleibt bei seiner ursprünglichen Drehung, wenn der Füllstand kleiner ist als startTurningAt (X Y Z).
„Zahl Zahl Zahl“	Gibt die Drehung des Shapes an, wenn der Füllstand kleiner ist als startTurningAt (X Y Z).

### 2.11.15. higherRotation (string)

Ausprägung	Beschreibung
highPosition (default)	Das Shape verbleibt bei highRotation, wenn der Füllstand größer ist als stopTurningAt (X Y Z).
„Zahl Zahl Zahl“	Gibt die Drehung des Shapes an, wenn der Füllstand größer ist als stopRotationAt (X Y Z).

### 2.11.16. turningType (string)

Art die Drehung zwischen lowRotation und highRotation in Relation zur Füllstandsänderung zurückzulegen.

Ausprägung	Beschreibung
„linear“ (default)	linear
„sphere“	wie eine Halbkugel
„cone“	wie ein Kegel
„square“	quadratisch
„circle“	wie ein Kreis
„sinus“	anhand einer Sinus-Funktion

### 2.11.17. startVisibilityAt (float)

Ausprägung	Beschreibung
-1 (default)	Das Shape wird ab 0 Litern angezeigt.
(Zahl)	Das Shape wird ab X Litern angezeigt. Die Bedingung wird mit „größer als“ (nicht „größer gleich“) überprüft.

### 2.11.18. stopVisibilityAt (float)

Ausprägung	Beschreibung
größte Kapazität der Fülltypen +1 (default)	Das Shape wird bis zur Maximalfüllmenge angezeigt.
(Zahl)	Das Shape wird bis X Litern Füllstand angezeigt. Die Bedingung wird mit „kleiner als“ (nicht „kleiner gleich“) überprüft.

## 2.12. parktrigger

Dieser Trigger deaktiviert die Auswahl eines Fahrzeuges per Tab, wenn es in ihm ist.

Die Default-Werte der Trigger-UserAttributes sind folgende:

UserAttribute	Default-Wert
allowMotorized	true
(andere)	false

## 2.13. processor

Das Kernstück des UPK. Es erzeugt Güter bzw. Früchte, wenn gewünscht verbraucht es dafür andere.

Der Referenzwert ist 1 Liter des product (siehe), zu dem alle anderen Mengenangaben im Verhältnis stehen.

### 2.13.1. **product (string)**

Legt den Fülltyp der Produktion fest.

Bsp: „wheat“, „barley“

### 2.13.2. **recipe (string)**

Ausprägung	Beschreibung
(ohne) (default)	Keine Füllstände werden bei der Produktion verbraucht.
(Auflistung von Füllständen und Fülltypen)	Legt die verbrauchte Mengen an Fülltypen fest, die für 1 Liter des Produkts verbraucht werden sollen. Beeinflusst die produzierte Menge in einem Produktionszyklus. Bsp: „2 water“, „5 manure 100 barley 30 water“

### 2.13.3. **byproducts (string)**

Gibt an, welche Mengen von anderen Fruchttypen noch produziert werden sollen.

Ausprägung	Beschreibung
(ohne) (default)	Es wird neben dem Produkt nichts weiter produziert.
(Auflistung von Füllständen und Fülltypen)	Zusätzlich und in Relation zu einem Liter des Produkts hergestellte Füllmengen. Bsp: „2 water“, „5 manure 100 barley 30 water“

### 2.13.4. **onlyWholeProducts (boolean)**

Ob das angegebene Produkt nur als Ganzes produziert werden soll. Die byproducts sind davon nicht betroffen.

Ausprägung	Beschreibung
true	Das Produkt wird nur in ganzzahliger Stückmenge produziert. Anfallende Reste werden beim nächsten Produktionszyklus dazugerechnet. Z. Bsp. für alle Dinge wie Tiere, Flaschen, Ballen etc.
false (default)	Das Produkt wird auch mit Nachkommasetellen dem Füllstand dazugerechnet. Z. Bsp. für alle Volumenangaben wie Weizen, Wasser, Dünger etc.



### 2.13.5. productionPrerequisite (string)

Legt die Voraussetzungen fest, wann 1 Liter des Produkts produziert werden kann.

Ausprägung	Beschreibung
<i>(ohne)</i> (default)	Diese Bedingung wird nicht angewandt.
<i>(Auflistung von Füllständen und Fülltypen)</i>	Zusätzlich und in Relation zu einem Liter des Produkts benötigte Füllmengen. Diese Füllstände werden nicht verbraucht. Beeinflusst die produzierte Menge in einem Produktionszyklus. Bsp: „2 water“, „5 manure 100 barley 30 water“

### 2.13.6. productsPerDay (float)

Ausprägung	Beschreibung
0 (default)	nichts wird beim Tageswechsel produziert.
<i>(Zahl)</i>	Legt die Menge des Produkt fest, die beim Tageswechsel produziert werden soll. Beeinflusst, ob ein Produktionszyklus ausgeführt wird. Nicht kombinierbar mit productsPerHour, productsPerMinute und productsPerSecond

### 2.13.7. productsPerHour (float)

Ausprägung	Beschreibung
0 (default)	nichts wird beim Stundenwechsel produziert.
<i>(Zahl)</i>	Legt die Menge des Produkt fest, die beim Stundenwechsel produziert werden soll. Beeinflusst, ob ein Produktionszyklus ausgeführt wird. Nicht kombinierbar mit productsPerDay, productsPerMinute und productsPerSecond

### 2.13.8. productsPerMinute (float)

Ausprägung	Beschreibung
0 (default)	nichts wird beim Minutenwechsel produziert.
<i>(Zahl)</i>	Legt die Menge des Produkt fest, die beim Minutenwechsel produziert werden soll. Beeinflusst, ob ein Produktionszyklus ausgeführt wird. Nicht kombinierbar mit productsPerDay, productsPerHour und productsPerSecond

### 2.13.9. productsPerSecond (float)

Ausprägung	Beschreibung
0 (default)	nichts wird beim Sekundenwechsel produziert.
(Zahl)	Legt die Menge des Produkt fest, die jede Sekunde (Echtzeit) produziert werden soll. Beeinflusst, ob ein Produktionszyklus ausgeführt wird. Nicht kombinierbar mit productsPerDay, productsPerHour und productsPerMinute

### 2.13.10. outcomeVariation (float)

Ausprägung	Beschreibung
0 (default)	Die produzierte Füllmenge schwankt nicht zufällig.
(Kommazahl zwischen 0 und 1)	Legt den Anteil fest, um den die Produktion schwanken soll. Beeinflusst die produzierte Menge in einem Produktionszyklus. Z.Bsp. „0.05“ entspricht 5% Schwankung nach oben und unten

### 2.13.11. outcomeVariationType (string)

Legt das Zufallsverfahren für die Produktionsschwankung fest.  
Nur in Kombination mit outcomeVariation.

Ausprägung	Beschreibung
uniform (default)	Die Produktionsschwankung ist gleichverteilt.
normal	Die Produktionsschwankung ist normalverteilt.

### 2.13.12. productionHours (string)

Zu welchen Uhrzeiten produziert werden soll.  
Nicht in Kombination mit productionPerDay.

Ausprägung	Beschreibung
0-23 (default)	Zu jeder Uhrzeit wird produziert
(Zahl-Zahl, Zahl-Zahl)	Legt die Uhrzeiten in Stunden fest, in denen produziert werden soll. Beeinflusst, ob ein Produktionszyklus ausgeführt wird. Bsp: „7-18“, „7-12, 14-16, 18-21“

### 2.13.13. productionInterval (float)

Ausprägung	Beschreibung
1 (default)	In jedem Zyklus wird produziert.
(Zahl)	Legt die Produktionszyklen fest, bei denen produziert werden soll. Zu lesen als „jeder x-ter Zyklus“. Beeinflusst, ob ein Produktionszyklus ausgeführt wird. Bsp: „2“, „5“

#### 2.13.14. productionProbability (float)

Ausprägung	Beschreibung
1 (default)	Es wird immer produziert
(Kommazahl zwischen 0 und 1)	Legt die Wahrscheinlichkeit fest, ob produziert werden soll. Beeinflusst, ob ein Produktionszyklus ausgeführt wird. Bsp: „0.5“, „0.2“ heißt, dass in 50% bzw. 20% der Fälle produziert wird.

#### 2.13.15. addIfProcessing (boolean)

Ausprägung	Beschreibung
(ohne) (default)	Es wird nichts hinzugefügt
(Auflistung von Füllständen und Fülltypen)	Fügt bestimmte Füllstände hinzu, wenn bei einem Produktionszyklus etwas produziert wurde. Im Unterschied zu product oder byproducts ist diese hinzugefügte Füllmenge von der Menge der produzierten Güter unabhängig. Bsp: „1 counter“, „5 manure 100 barley 30 water“

#### 2.13.16. addIfNotProcessing (boolean)

Ausprägung	Beschreibung
(ohne) (default)	Es wird nichts hinzugefügt
(Auflistung von Füllständen und Fülltypen)	Fügt bestimmte Füllstände hinzu, wenn bei einem Produktionszyklus nichts produziert wurde (z.B. weil productionPrerequisite (siehe 2.13.5) nicht erfüllt wurde). Bsp: „1 counter“, „5 manure 100 barley 30 water“

#### 2.13.17. emptyFillTypesIfProcessing (boolean)

Ausprägung	Beschreibung
(ohne) (default)	Es wird nichts hinzugefügt
(Auflistung von Fülltypen)	Leert bestimmte Füllstände, dh. setzt sie auf 0, wenn etwas produziert wird. Bspw. um einen Zähler der unproduktiven Zyklen zurückzusetzen. Bsp: „counter“, „wheat barley“

### 2.13.18. emptyFillTypesIfNotProcessing (boolean)

Ausprägung	Beschreibung
(ohne) (default)	Es wird nichts hinzugefügt
(Auflistung von Fülltypen)	Leert bestimmte Füllstände, dh. setzt sie auf 0, wenn nichts produziert wird. Bspw. um einen Zähler der produktiven Zyklen zurückzusetzen. Bsp: „counter“, „wheat barley“

### 2.13.19. enableChildrenIfProcessing (boolean)

Ausprägung	Beschreibung
true	Die untergeordneten Module werden aktiviert, wenn etwas produziert wird, um bspw. eine Animation abzuspielen. Nicht in Kombination mit disableChildrenIfProcessing.
false (default)	Die untergeordneten Module werden weder aktiviert noch deaktiviert.

### 2.13.20. enableChildrenIfNotProcessing (boolean)

Ausprägung	Beschreibung
true	Die untergeordneten Module werden aktiviert, wenn in einem Produktionszyklus nichts produziert wird. Nicht in Kombination mit disableChildrenIfNotProcessing.
false (default)	Die untergeordneten Module werden weder aktiviert noch deaktiviert.

### 2.13.21. disableChildrenIfProcessing (boolean)

Ausprägung	Beschreibung
true	Die untergeordneten Module werden deaktiviert, wenn etwas produziert wird, um bspw. eine Animation abzuspielen. Nicht in Kombination mit enableChildrenIfProcessing.
false (default)	Die untergeordneten Module werden weder aktiviert noch deaktiviert.

### 2.13.22. disableChildrenIfNotProcessing (boolean)

Ausprägung	Beschreibung
true	Die untergeordneten Module werden deaktiviert, wenn in einem Produktionszyklus nichts produziert wird. Nicht in Kombination mit enableChildrenIfNotProcessing.
false (default)	Die untergeordneten Module werden weder aktiviert noch deaktiviert.

## **2.14. selltarget**

Dieses Modul richtet eine (transparente) Plane beim Verkaufen von platzierbaren Mods so aus, dass sie auf Klicks reagiert und schließlich der Mod verkauft werden kann. Denn manchmal kann es vorkommen, dass ein Mod nur aus (transparenten) Shapes besteht, die nicht auf die Klicks beim Verkaufen reagieren. Oder der Mod ist sehr klein und schwer anzuklicken.

## 2.15. sprayerfilltrigger

Dieser Trigger befüllt Spritzen und Düngestreuer mit Dünger. Im Gegensatz zum filltrigger muss die Abfüllung vom Fahrzeug aus bestätigt werden, genauso wie die Standard-Trigger.

Die Default-Werte der Trigger-UserAttributes sind folgende:

UserAttribute	Default-Wert
allowSprayer	true
(andere)	false

### 2.15.1. createFillType (boolean)

Ausprägung	Beschreibung
true	Legt fest, dass die Befüllung unabhängig vom Füllstand stattfindet.
false (default)	Legt fest, dass die abgefüllte Füllmenge vorhanden sein muss.

### 2.15.2. pricePerLiter (float)

Ausprägung	Beschreibung
0 (default)	Die Befüllung ist kostenlos.
(Zahl)	Die Befüllung kostet X € pro Liter. Negative Werte werden zum Kontostand hinzugefügt. Mit statName kombinierbar.

### 2.15.3. statName (string)

Falls die Abfüllung Geld kostet (oder einbringt), legt statName fest, auf welchen Punkt in der Spiel-Statistik der Betrag gebucht werden soll.

Ausprägung	Beschreibung
„other“ (default)	Sonstiges
„newVehiclesCost“	Neue Fahrzeuge
„newAnimalsCost“	Tierkosten
„constructionCost“	Baukosten
„vehicleRunningCost“	Betriebskosten Fahrzeuge
„propertyMaintenance“	Betriebskosten Gebäude
„wagePayment“	Lohnzahlungen
„harvestIncome“	Einnahmen Ernte
„missionIncome“	Einnahmen Missionen
„loanInterest“	Kreditzinsen

## 2.16. switcher

Dieses Modul tauscht je nach Füllstand oder Fülltyp Shapes gegeneinander aus. Die dafür benutzten Shapes sind die untergeordneten des Moduls.

Um, wenn gewünscht, bei einem bestimmten Füllstandsabschnitt nichts angezeigt werden soll, sind auch TransformGroups statt Shapes möglich.

### 2.16.1. **fillTypes (string)**

Ausprägung	Beschreibung
<i>(ohne)</i> (default)	Auf nichts wird reagiert.
<i>(Auflistung von Fülltypen)</i>	Die Fülltypen, auf die reagiert werden soll. Bsp: „wheat“, „wheat barley“

### 2.16.2. **fillTypeChoice (string)**

Ausprägung	Beschreibung
„max“ (default)	Wählt den Füllstand desjenigen Fülltyps aus, der am höchsten ist.
„min“	Wählt den Füllstand desjenigen Fülltyps aus, der am niedrigsten ist.

### 2.16.3. **switchFillTypes (string)**

Falls auf den Fülltyp reagiert werden soll. Nur sinnvoll mit den Speicherarten single, fifo oder filo verwendbar.

Ausprägung	Beschreibung
<i>(ohne)</i> (default)	Auf nichts wird reagiert.
<i>(Auflistung von Fülltypen, mit Komma getrennt)</i>	Ordnet der Reihenfolge nach bestimmte Fülltypen den Shapes zu. Mehrere Fülltypen pro Shape können mit Leerzeichen getrennt aufgeführt werden. Bsp: „wheat barley, rape, maize“

### 2.16.4. **switchFillLevels**

Falls auf den Füllstand reagiert werden soll.

Ausprägung	Beschreibung
<i>(ohne)</i> (default)	Auf nichts wird reagiert.
<i>(Auflistung von Füllständen)</i>	Ordnet der Reihenfolge nach bestimmte maximale Füllstände den Shapes zu. Das erste Shape wird bis zum ersten angegebenen Füllstand angezeigt, das letzte Shape ab dem zuletzt aufgeführten Füllstand. Bsp: „1500 3000“

### 2.16.5. mode (string)

Gibt den Modus für switchFillLevels an.

Ausprägung	Beschreibung
„switch“ (default)	Die Shapes werden je nach Füllstand nach und nach gegeneinander ausgetauscht.
„stack“	Die Shapes werden alle bis zu dem aktuellen Füllstand angezeigt.
„stackReverse“	<i>in Bearbeitung</i>

### 2.16.6. hidingPosition (string)

Ausprägung	Beschreibung
„0 -10 0“ (default)	Das Shape wird, wenn es ausgeblendet ist, 10m nach unten verschoben, um ev. Kollisionen zu vermeiden.
(Zahl Zahl Zahl)	Auf welche relative Position das Shape verschoben werden soll, wenn es ausgeblendet ist.



## 2.17. tiptrigger

Dieser Trigger ermöglicht es Kipper zu entladen und deren Kipp-Animation abzuspielen.

Die Default-Werte der Trigger-UserAttributes sind folgende:

UserAttribute	Default-Wert
allowTipper	true
(andere)	false

### 2.17.1. **acceptedFillTypes (string)**

Ausprägung	Beschreibung
(ohne) (default)	Es wird nichts akzeptiert
(Auflistung von Fülltypen)	Akzeptiert bestimmte Füllsorten Bsp: „rape“, „wheat barley“

### 2.17.2. **showNotAcceptedWarning (boolean)**

Ausprägung	Beschreibung
true (default)	Es wird eine Warnung angezeigt, wenn der Fülltyp des Kippers vom tiptrigger nicht akzeptiert wird. Der angezeigte Text kann im l10n-Abschnitt der modDesc.xml mit „notAcceptedHere“ angepasst werden.
false	Es wird keine Warnung angezeigt.

### 2.17.3. **showCapacityReachedWarning (boolean)**

Ausprägung	Beschreibung
true (default)	Es wird eine Warnung angezeigt, wenn die maximale Füllmenge der Füllsorte bereits erreicht ist. Der angezeigte Text kann im l10n-Abschnitt der modDesc.xml mit „capacityReached“ angepasst werden.
false	Es wird keine Warnung angezeigt.

### 2.17.4. **revenuePerLiter (float)**

revenuePerLiter gilt für alle erlaubten Fruchttypen. Davon kann man mit revenuesPerLiter Ausnahmen festlegen bzw. gleich alle mit Preisen auflisten. Zuerst schaut das Skript ob in revenuesPerLiter ein Wert für den Fruchttyp festgelegt wurde, sonst nimmt es den Wert von revenuePerLiter (per default auf 0€ pro Liter gesetzt).

ZBsp.: acceptedFillTypes="wheat barley maize rape", revenuePerLiter="0.1", revenuesPerLiter="1.2 wheat 1.3 barley" und convertFillTypes="void wheat rape".  
Für Weizen bekommt man 1,2€ pro Liter und es wird gelöscht. Für Barley bekommt

man 1,3€ pro Liter und es wird gespeichert. Für Mais bekommt man 0,1€ und es wird gespeichert. Für Raps bekommt man auch 0,1€ und es wird gelöscht.

Ausprägung	Beschreibung
0 (default)	Die Entleerung ist kostenlos.
(Zahl)	Die Entleerung kostet X € pro Liter. Negative Werte werden zum Kontostand hinzugefügt. Mit statName kombinierbar.

#### 2.17.5. revenuesPerLiter (string)

Ausprägung	Beschreibung
(ohne) (default)	Die Entleerung ist kostenlos bzw. es gilt pricePerLiter.
(Auflistung von Kosten und Fülltypen)	Die Entleerung kostet X € pro Liter. Negative Werte werden dem Kontostand hinzugefügt. Mit statName kombinierbar. Für nicht hier aufgeführte, aber im Trigger akzeptierte Fülltypen, gilt pricePerLiter. Bsp: „1.2 wheat 1.3 barley“

#### 2.17.6. statName (string)

Falls die Abfüllung Geld kostet (oder einbringt), legt statName fest, auf welchen Punkt in der Spiel-Statistik der Betrag gebucht werden soll.

Ausprägung	Beschreibung
„other“ (default)	Sonstiges
„newVehiclesCost“	Neue Fahrzeuge
„newAnimalsCost“	Tierkosten
„constructionCost“	Baukosten
„vehicleRunningCost“	Betriebskosten Fahrzeuge
„propertyMaintenance“	Betriebskosten Gebäude
„wagePayment“	Lohnzahlungen
„harvestIncome“	Einnahmen Ernte
„missionIncome“	Einnahmen Missionen
„loanInterest“	Kreditzinsen

spezielle UserAttributes - unspecified

## **2.18. unspecified**

Dieses Modul unterstützt nur die Standard-Attribute (siehe 1.1). Hauptsächlich zur Speicherung von Füllständen und Verbindung von Modulen gedacht.

## 2.19. waterfilltrigger

Dieser Trigger befüllt Wasseranhänger mit Wasser. Im Gegensatz zum filltrigger muss die Abfüllung vom Fahrzeug aus bestätigt werden, genauso wie die Standard-Trigger.

Die Default-Werte der Trigger-UserAttributes sind folgende:

UserAttribute	Default-Wert
allowWaterTrailer	true
(andere)	false

### 2.19.1. createFillType (boolean)

Ausprägung	Beschreibung
true	Legt fest, dass die Befüllung unabhängig vom Füllstand stattfindet.
false (default)	Legt fest, dass die abgefüllte Füllmenge vorhanden sein muss.

### 2.19.2. pricePerLiter (float)

Ausprägung	Beschreibung
0 (default)	Die Befüllung ist kostenlos.
(Zahl)	Die Befüllung kostet X € pro Liter. Negative Werte werden zum Kontostand hinzugefügt. Mit statName kombinierbar.

### 2.19.3. statName (string)

Falls die Abfüllung Geld kostet (oder einbringt), legt statName fest, auf welchen Punkt in der Spiel-Statistik der Betrag gebucht werden soll.

Ausprägung	Beschreibung
„other“ (default)	Sonstiges
„newVehiclesCost“	Neue Fahrzeuge
„newAnimalsCost“	Tierkosten
„constructionCost“	Baukosten
„vehicleRunningCost“	Betriebskosten Fahrzeuge
„propertyMaintenance“	Betriebskosten Gebäude
„wagePayment“	Lohnzahlungen
„harvestIncome“	Einnahmen Ernte
„missionIncome“	Einnahmen Missionen
„loanInterest“	Kreditzinsen