Attach Implements Manually

Einbauanleitung / Setup Guide

Contents

1 German/Deutsch **1**

1.1 Anpassung der Triggergröße für Fahrzeuge . . . 1
1.1.1 Einleitung . . . 1
1.1.2 Anpassung der Triggergröße . . . 1
1.2 Anpassung der Kupplungstypen bei Frontladern . . . 4
1.2.1 Einleitung . . . 4
1.2.2 Anpassung der Kupplungstypen . . . 5
1.3 Optionen für InputAttacherJoints . . . 6
1.3.1 Einleitung . . . 6
1.3.2 Anpassung des InputAttacherJoints . . . 6
1.4 Schlauchstatus bei Hakenliften . . . 7
1.5 Handbremse und Unterlegkeile . . . 7
1.5.1 Einleitung . . . 7
1.5.2 Anpassung der Handbremse . . . 8
1.5.3 Unterlegkeile . . . 10
1.6 Globale Konfigurationsdatei für Attach Implements Manually . . . 10
1.6.1 Einleitung . . . 10
1.6.2 Aufbau der Konfigurationsdatei . . . 10
1.6.3 Eigene Frontlader-AttacherJoint-Typen . . . 13
1.6.4 Fahrzeuge deaktivieren . . . 14
1.6.5 Eigene Schlauchgruppen . . . 14
1.6.6 Automatisch gekuppelte eigene Schläuche . . . 15
1.6.7 Handbremse überschreiben . . . 15

2 English **16**

2.1 Adjusting Trigger Size for Vehicles . . . 16
2.1.1 Introduction . . . 16
2.1.2 Adjusting the Trigger Size . . . 16
2.2 Adjusting Coupling Types for Front Loaders . . . 19
2.2.1 Introduction . . . 19
2.2.2 Adjusting the Coupling Types . . . 20
2.3 Input Attacher Joint Options . . . 21
2.3.1 Introduction . . . 21
2.3.2 Adjusting the Input Attacher Joint . . . 21
2.4 Hose Status of Hook Lifts . . . 22
2.5 Handbrake and Wheel Chocks . . . 22

© 2022 - 2026 BayernGamers

2.5.1 Introduction 22
2.5.2 Adjusting the Handbrake 23
2.5.3 Wheel Chocks 24
2.6 Global Configuration File for Attach Implements Manually 25
2.6.1 Introduction 25
2.6.2 Configuration File Structure 25
2.6.3 Custom Front Loader Attacher Joint Types 28
2.6.4 Disabling Vehicles 29
2.6.5 Custom Hose Groups 29
2.6.6 Automatically Coupled Custom Hoses 29
2.6.7 Overriding the Handbrake 30

© 2022 - 2026 BayernGamers

1 German/Deutsch

1.1 Anpassung der Triggergröße für Fahrzeuge

1.1.1 Einleitung

Dieser Abschnitt beschreibt die Anpassung der Triggergröße bei Fahrzeugen. Dies kann manchmal erforderlich sein, wenn die in der XML definierte Fahrzeuggröße nicht zu den eigentlichen Abmessungen des Fahrzeugs passt oder wenn der Trigger an einer bestimmten Position des Fahrzeugs und ggf. auch erheblich kleiner oder größer als das Fahrzeug selbst sein soll. Hierbei gilt es zu beachten, dass jeweils nur ein Trigger pro Fahrzeug erstellt wird.

Ab Version 1.0.1.0 ist der Trigger kein physikalischer Trigger mehr, sondern ein Erkennungsbereich in Form einer Box, in dem die Position des Spielers geprüft wird. Dieser Bereich wird auch für die Handbremse und die Unterlegkeile von Anhängern verwendet. Die XML-Attribute bleiben unverändert.

1.1.2 Anpassung der Triggergröße

Die Triggergröße eines Fahrzeugs wird standardmäßig durch die in der XML definierten Fahrzeugabmessungen bestimmt. Diese sollten idealerweise das gesamte Fahrzeug in seiner Höhe, Breite und Länge umfassen. Die Fahrzeugabmessungen können im Spiel durch den Konsolenbefehl `gsVehicleDebug` sichtbar gemacht werden. Danach erscheint am unteren Rand des Bildschirms eine Art Menüleiste. Hier ist für uns der Punkt `Attributes` relevant. Diesen wählt man anschließend über `Shift + 5` aus. Daraufhin erscheinen einige Boxen und Texte rund um das aktuelle Fahrzeug. Dabei stellt die dunkelblaue Box die Abmessungen des Fahrzeugs dar. Abbildung 1 zeigt die Anzeige der Fahrzeugabmessungen durch den Konsolenbefehl `gsVehicleDebug`.

© 2022 - 2026 BayernGamers



Figure 1: Anzeige der Fahrzeugabmessungen durch den Konsolenbefehl `gsVehicleDebug`

Die Werte der Fahrzeugabmessungen werden dabei in der Fahrzeug-XML über die Attribute `length`, `width` und `height` definiert. Zusätzlich kann ein Offset über die Attribute `lengthOffset`,

© 2022 - 2026 BayernGamers

`widthOffset` und `heightOffset` definiert werden. Diese Werte verschieben den Trigger relativ zum Fahrzeugmittelpunkt.

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<vehicle type="tractor" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
    noNamespaceSchemaLocation="../../../../shared/xml/schema/vehicle.xsd">
    <annotation>Copyright (C) GIANTS Software GmbH, All Rights Reserved.</annotation
        >

    <storeData>
        [...]
    </storeData>

    <base>
        [...]
        <!-- Definition der Fahrzeugabmessungen -->
        <size width="2.75" length="5.2" height="3.2" lengthOffset="0.35"/>
        [...]
    </base>
    [...]
</vehicle>
```

Die Fahrzeugabmessungen ergeben anschließend die Größe des Triggers, den Attach Implements Manually zur Erkennung von Spielern verwendet. Dabei wird die Länge des Fahrzeugs mit dem Faktor 1.2 und die Breite des Fahrzeugs mit dem Faktor 1.3 multipliziert. Der Faktor für die Breite wird zudem auf 2.3 erhöht, wenn die Breite des Fahrzeugs kleiner als 2 Meter ist. Ab Version 1.0.1.0 werden dabei auch Konfigurationen berücksichtigt, die die Fahrzeugabmessungen verändern, zum Beispiel eine andere Länge eines Anhängers.

Um die Abmessungen des Triggers zu verändern, kann in der Fahrzeug-XML der folgende Eintrag hinzugefügt werden:

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<vehicle type="tractor" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
    noNamespaceSchemaLocation="../../../../shared/xml/schema/vehicle.xsd">
    <annotation>Copyright (C) GIANTS Software GmbH, All Rights Reserved.</annotation
        >

    <storeData>
        [...]
    </storeData>

    <base>
        [...]
    </base>
    [...]
    <attachImplementsManually>
```


© 2022 - 2026 BayernGamers

```
14          <trigger rootNode="0>1|0" offset="0 1.5 0" size="3 3.5 6"/>
15      </attachImplementsManually>
16  </vehicle>
```

Das Attribut `rootNode` definiert, an welcher Transformgruppe sich der Trigger ausrichtet. Ohne Angabe wird der Root-Node des Fahrzeugs verwendet. Die Attribute `offset` und `size` definieren die Position und Größe des Triggers.

`size` gibt die Breite, Höhe und Länge des Triggers in Metern an. Die Werte ersetzen die aus den Fahrzeugabmessungen berechnete Größe vollständig, die oben genannten Faktoren werden dann nicht angewendet. Im gezeigten Beispiel ist der Trigger also 3 Meter breit, 3,5 Meter hoch und 6 Meter lang.

`offset` gibt die Position der Triggermitte relativ zum `rootNode` in Metern an (x, y, z). Er überschreibt die Offset-Werte, die Attach Implements Manually standardmäßig aus den Fahrzeugabmessungen ausliest. Im gezeigten Beispiel liegt die Mitte des Triggers 1,5 Meter über dem `rootNode`.

Bis Version 1.0.0.4 beschrieb diese Anleitung `size` als Skalierung relativ zu den Fahrzeugabmessungen. Das war falsch: Der Mod hat die Werte schon immer als Abmessungen in Metern verwendet. Bestehende Einträge verhalten sich daher unverändert, sollten aber mit der Beschreibung oben geprüft werden.

1.2 Anpassung der Kupplungstypen bei Frontladern

1.2.1 Einleitung

Attach Implements Manually fügt Frontladern, Teleskopladern, Radladern und Kompaktladern automatisch eine neue Konfigurationsmöglichkeit hinzu, welche es dem Spieler ermöglicht, zwischen drei verschiedenen Kupplungstypen zu wechseln. Hierbei kann der Spieler zwischen den Kupplungstypen `MANUAL`, `SEMI_AUTOMATIC` und `AUTOMATIC` wechseln. Um Moddern die Möglichkeit zu geben, die verfügbaren Kupplungstypen für ein Fahrzeug einzuschränken, bietet Attach Implements Manually die Möglichkeit, die verfügbaren Kupplungstypen über die Fahrzeug-XML zu definieren. Hierbei kann auch ein Standard-Kupplungstyp definiert werden, welcher beim Kauf des Fahrzeugs vorausgewählt ist.

© 2022 - 2026 BayernGamers

1.2.2 Anpassung der Kupplungstypen

Kupplungstyp	Beschreibung	Anwendungsfall
MANUAL	Der Spieler muss die Kupplung und die Schläuche manuell anbringen.	Für Fahrzeuge, bei denen in der Realität die Verriegelung des Anbaugeräts manuell erfolgen muss. Schläuche müssen manuell verbunden werden.
SEMI_AUTO-MATIC	Der Spieler kann ein Anbaugerät vom Fahrzeug aus mit `Q` ankuppeln. Schläuche müssen manuell verbunden werden.	Für Fahrzeuge, die in der Realität über eine automatisch verriegelnde oder von innen bedienbare Verriegelung verfügen. Schläuche müssen weiterhin manuell verbunden werden.
AUTOMATIC	Der Spieler kann ein Anbaugerät vom Fahrzeug aus mit `Q` ankuppeln. Schläuche werden automatisch verbunden.	Für Fahrzeuge, die in der Realität über eine automatisch verriegelnde oder von innen bedienbare Verriegelung verfügen und einen Hydraulik-Schnellkuppler besitzen.

Table 1: Übersicht der verschiedenen Kupplungstypen

Standardmäßig sind alle Kupplungstypen aktiviert, sodass der Spieler frei wählen kann, welchen Kupplungstyp er verwenden möchte. Es ist jedoch auch möglich, die verfügbaren Kupplungstypen für ein Fahrzeug einzuschränken. Hierfür muss in der Fahrzeug-XML der folgende Eintrag hinzugefügt werden:

```
1 <?xml version="1.0" encoding="utf-8" standalone="no"?>
2 <vehicle type="tractor" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
      noNamespaceSchemaLocation="../../../../shared/xml/schema/vehicle.xsd">
3     <annotation>Copyright (C) GIANTS Software GmbH, All Rights Reserved.</annotation
          >
4
5     <storeData>
6         [...]
7     </storeData>
8
9     <base>
```


© 2022 - 2026 BayernGamers

```
10          [...]
11      </base>
12      [...]
13      <attachImplementsManually>
14          <defaultFrontloaderAttacherType type="SEMI_AUTOMATIC" allowSelection="true"
        allowedTypes="SEMI_AUTOMATIC␣AUTOMATIC"/>
15      </attachImplementsManually>
16  </vehicle>
```

Das gezeigte Beispiel ermöglicht es dem Spieler, zwischen den Kupplungstypen `SEMI_AUTOMATIC` und `AUTOMATIC` zu wählen. Der Kupplungstyp `MANUAL` ist in diesem Fall nicht verfügbar. Das Attribut `defaultFrontloaderAttacherType` definiert den standardmäßig ausgewählten Kupplungstyp beim Kauf des Fahrzeugs. Das Attribut `allowSelection` definiert, ob der Spieler die Möglichkeit haben soll, zwischen verschiedenen Kupplungstypen zu wählen. Das Attribut `allowedTypes` definiert schließlich, welche Kupplungstypen zur Auswahl stehen sollen. Ist `allowSelection` auf `false` gesetzt, so wird der Kupplungstyp, der in `defaultFrontloaderAttacherType` definiert ist, fest für das Fahrzeug eingestellt und kann nicht geändert werden.

1.3 Optionen für InputAttacherJoints

1.3.1 Einleitung

Attach Implements Manually ergänzt jeden `inputAttacherJoint` eines Geräts um einige optionale Attribute. Sie werden sowohl in `vehicle.attachable.inputAttacherJoints` als auch in `inputAttacherJointConfigurations` gelesen. Damit kann ein Gerät automatisch ankuppeln oder aus der Kabine angekuppelt werden, ohne dass die globalen Einstellungen geändert werden müssen.

1.3.2 Anpassung des InputAttacherJoints

```
1   <vehicle type="trailer">
2       [...]
3       <attachable>
4           <inputAttacherJoints>
5               <inputAttacherJoint node="attacherJoint" jointType="trailer" autoAttach=
        "true" autoAttachDistance="0.15" allowAttachFromInside="false"
        attachHosesIfAttachedFromInside="false" attachPTOIfAttachedFromInside="false"/>
6           </inputAttacherJoints>
7       </attachable>
8       [...]
9   </vehicle>
```


© 2022 - 2026 BayernGamers

Attribut	Typ / Standard	Beschreibung
`autoAttach`	Bool, `false`	Das Gerät kuppelt automatisch an, sobald der AttacherJoint des Fahrzeugs nah genug ist, wie bei Anhängern mit Zugmaul.
`autoAttachDistance`	Float, `0.1`	Abstand in Metern für `autoAttach`. Werte unter `0.001` werden auf `0.001` angehoben.
`allowAttachFromInside`	Bool, `false`	Das Gerät kann wie ohne Attach Implements Manually aus der Kabine mit `Q` angekuppelt werden.
`attachHosesIf AttachedFromInside`	Bool, `false`	Die Schläuche werden automatisch verbunden, wenn das Gerät aus der Kabine angekuppelt wird.
`attachPTOIf AttachedFromInside`	Bool, `false`	Die Zapfwelle wird automatisch verbunden, wenn das Gerät aus der Kabine angekuppelt wird.

Table 2: InputAttacherJoint-Attribute von Attach Implements Manually

1.4 Schlauchstatus bei Hakenliften

Bei einem Hakenlift werden die Schläuche zu einem Abrollcontainer nur angeboten, solange der Hakenlift ausgeklappt ist. Manche Hakenlift-Fahrzeuge verwenden einen invertierten Klappzustand. Für diese Fahrzeuge kann der Schlauchstatus in der XML des Hakenlift-Fahrzeugs invertiert werden:

```
<vehicle type="hookLiftTrailer">
    [...]
    <attachImplementsManually>
        <hookLift invertHoseStatus="true"/>
    </attachImplementsManually>
</vehicle>
```

1.5 Handbremse und Unterlegkeile

1.5.1 Einleitung

Attach Implements Manually ergänzt die Shop-Konfiguration `handbrakeType` mit den Typen `NONE`, `HANDBRAKE` und `SPRING_BRAKE`. Die Konfiguration wird automatisch jedem gezogenen

© 2022 - 2026 BayernGamers

Fahrzeug mit Rädern hinzugefügt. Ein Fahrzeug gilt als gezogen, wenn einer seiner InputAttacherJoints einen der Joint-Typen `trailer`, `trailerLow`, `trailerCar`, `trailerSaddled`, `semitrailer`, `semitrailerHook`, `semitrailerCar`, `cutterTrailer` oder `hookLift` verwendet. Sämaschinen mit Druckluftanlage erhalten die Konfiguration ebenfalls.

Ohne Eintrag in der Fahrzeug-XML wird der Standardtyp automatisch erkannt:

- `SPRING_BRAKE`, wenn das Fahrzeug eine Druckluftanlage hat (`attachable.airConsumer#usage` größer 0 oder ein `airDoubleRed`- / `airDoubleYellow`-Schlauch oder -Target),
- sonst `HANDBRAKE`, wenn `attachable.brakeForce#force` größer 0 ist,
- sonst `NONE`.

`SPRING_BRAKE` ist nur für Fahrzeuge mit Druckluftanlage wählbar, außer es ist der Standardtyp.

1.5.2 Anpassung der Handbremse

```
1 <vehicle type="trailer">
2     [...]
3     <attachImplementsManually>
4         <handbrake type="SPRING_BRAKE" allowSelection="true" allowedTypes="HANDBRAKE
    ␣SPRING_BRAKE" animationName="handbrakeLever" useWheelChocks="true" force="2"
    pressureLossPerMinute="0.02"/>
5     </attachImplementsManually>
6 </vehicle>
```


© 2022 - 2026 BayernGamers

Attribut	Typ / Standard	Beschreibung
`type`	String	Standardtyp: `NONE`, `HANDBRAKE` oder `SPRING_BRAKE`. Ist er gesetzt, erhält das Fahrzeug die Konfiguration auch dann, wenn es nicht als gezogenes Fahrzeug erkannt wird.
`allowSelection`	Bool, `true`	Bei `false` ist im Shop nur der Standardtyp verfügbar.
`allowedTypes`	String-Liste	Typen, die der Spieler wählen kann. Ist die Liste leer, sind alle verfügbaren Typen wählbar.
`animationName`	String	Animation, die beim Anziehen der Handbremse abgespielt und beim Lösen rückwärts abgespielt wird.
`useWheelChocks`	Bool	Aktiviert oder deaktiviert die Aktion für die Unterlegkeile. Ohne Angabe ist die Aktion verfügbar, sobald die Räder des Fahrzeugs `wheelChock`-Einträge haben.
`force`	Float, `2`	Mindestbremskraft, solange Handbremse oder Unterlegkeile das Fahrzeug halten, in der Einheit von `attachable.brakeForce#force`. Wird für Fahrzeuge ohne eigene Bremskraft verwendet.
`pressureLossPerMinute`	Float, `0.02`	Druck in bar, den der Vorratsbehälter der Federspeicherbremse pro Minute verliert, solange die rote Druckluftleitung nicht verbunden ist.

Table 3: Attribute von `attachImplementsManually.handbrake`

Der Vorratsbehälter der Federspeicherbremse wird bei verbundener roter Druckluftleitung auf 8,5 bar gefüllt. Am Gerät lässt sich die Federspeicherbremse nur mit mindestens 5 bar Restdruck lösen. Jedes manuelle Lösen verbraucht 1 bar.

© 2022 - 2026 BayernGamers

1.5.3 Unterlegkeile

Die Unterlegkeile sind unabhängig vom Handbremstyp. Sie verwenden die `wheelChock`-Einträge der Räder aus dem Basisspiel. Fahrzeuge, die bereits Unterlegkeile haben, brauchen daher keinen weiteren Eintrag. Ein Fahrzeug mit angelegten Unterlegkeilen rollt nicht weg.

1.6 Globale Konfigurationsdatei für Attach Implements Manually

1.6.1 Einleitung

Zusätzlich zu den Einstellungen in der Fahrzeug-XML kann ein Mod eine globale Konfigurationsdatei für Attach Implements Manually bereitstellen. Diese Datei eignet sich für Einstellungen, die nicht direkt an einem einzelnen Fahrzeug-Eintrag liegen sollen. Darin können eigene Frontlader-AttacherJoint-Typen als Frontlader-Kupplungen markiert, eigene Schlauchtypen einer Schlauchgruppe zugeordnet, eigene Schlauchtypen automatisch gekuppelt, die Handbremsen-Ausstattung einzelner Fahrzeuge überschrieben und einzelne Fahrzeuge von Attach Implements Manually ausgeschlossen werden.

Die Datei kann frei benannt werden. Standardmäßig wird `attachImplementsManuallyConfig.xml` als Dateiname verwendet. Die Datei muss im Mod liegen und wird in der `modDesc.xml` über das `<attachImplementsManually>` Tag verlinkt:

```
1 <?xml version="1.0" encoding="utf-8" standalone="no"?>
2 <modDesc descVersion="109">
3     [...]
4     <attachImplementsManually configFilename="attachImplementsManuallyConfig.xml" />
5 </modDesc>
```

Der Pfad in `configFilename` ist relativ zum Stammverzeichnis des Mods. Liegt die Datei zum Beispiel in einem Unterordner `config`, kann entsprechend `config/attachImplementsManuallyConfig.xml` angegeben werden.

1.6.2 Aufbau der Konfigurationsdatei

Die Konfigurationsdatei verwendet `attachImplementsManuallyConfig` als Root-Element. Alle Einträge sind optional:

```
1 <?xml version="1.0" encoding="utf-8" standalone="no"?>
2 <attachImplementsManuallyConfig>
3     <customFrontloaderJointTypes>
4         <customFrontloaderJointType name="MX" />
5     </customFrontloaderJointTypes>
6
7     <customHoseGroups>
```


© 2022 - 2026 BayernGamers

```
8         <customHoseGroup name="myMultiCoupler" group="custom" provides="hydraulicIn␣
      hydraulicOut␣electric" />
9     </customHoseGroups>
10
11    <autoCoupleCustomHoseTypes>
12        <autoCoupleCustomHoseType name="myCamera*" />
13    </autoCoupleCustomHoseTypes>
14
15    <handbrakeVehicles>
16        <handbrakeVehicle xmlFilename="vehicles/myTrailer/myTrailer.xml" type="
      SPRING_BRAKE" />
17    </handbrakeVehicles>
18
19    <disabledVehicles>
20        <disabledVehicle xmlFilename="vehicles/myVehicle/myVehicle.xml" />
21    </disabledVehicles>
22 </attachImplementsManuallyConfig>
```

Element / Attribut	Typ	Beschreibung
`customFrontloader JointTypes`	Element	Container für eigene AttacherJoint-Typen, die Attach Implements Manually wie Frontlader-AttacherJoints behandeln soll.
`customFrontloaderJointType #name`	String	Name des eigenen Joint-Typs ohne den Prefix `JOINTTYPE_`. Beispiel: Für `name="MX"` wird intern `JOINTTYPE_MX` verwendet. Der Joint-Typ muss im Spiel bereits als AttacherJoint-Typ registriert sein.

Table 4: Parameter der `attachImplementsManuallyConfig.xml` (Teil 1)

© 2022 - 2026 BayernGamers

Element / Attribut	Typ	Beschreibung
`customHoseGroups`	Element	Container für eigene Schlauchtypen, die einer Schlauchgruppe zugeordnet werden.
`customHoseGroup #name`	String	Eigener Schlauchtyp (`type` eines `customHose` / `customTarget`). Groß-/Kleinschreibung wird ignoriert. Ein `*` am Ende trifft alle Typen mit diesem Präfix.
`customHoseGroup #group`	String	Schlauchgruppe, in der der Typ angeboten wird: `hydraulicIn`, `hydraulicOut`, `airDoubleRed`, `airDoubleYellow`, `electric`, `isobus`, `custom` oder `toolConnector`.
`customHoseGroup #provides`	String	Optional. Durch Leerzeichen getrennte Schlauchgruppen, die der Typ zusätzlich bereitstellt, zum Beispiel bei Multikupplungen.
`autoCoupleCustomHoseTypes`	Element	Container für eigene Schlauchtypen, die immer automatisch verbunden werden.
`autoCoupleCustomHoseType #name`	String	Eigener Schlauchtyp. Groß-/Kleinschreibung wird ignoriert. Ein `*` am Ende trifft alle Typen mit diesem Präfix.

Table 5: Parameter der `attachImplementsManuallyConfig.xml` (Teil 2)

© 2022 - 2026 BayernGamers

Element / Attribut	Typ	Beschreibung
`handbrakeVehicles`	Element	Container für Fahrzeuge, deren Handbremsen-Ausstattung überschrieben wird.
`handbrakeVehicle #xmlFilename`	String	Pfad zur Fahrzeug-XML, relativ zum Stammverzeichnis des Mods, der die Konfigurationsdatei bereitstellt.
`handbrakeVehicle #type`	String	Optional. `NONE`, `HANDBRAKE` oder `SPRING_BRAKE`.
`handbrakeVehicle #animationName, #useWheelChocks, #force`		Optional. Gleiche Bedeutung wie in `attachImplementsManually.handbrake` der Fahrzeug-XML.
`disabledVehicles`	Element	Container für Fahrzeuge, bei denen Attach Implements Manually deaktiviert werden soll.
`disabledVehicle #xmlFilename`	String	Pfad zur Fahrzeug-XML, relativ zum Stammverzeichnis des Mods, der die Konfigurationsdatei bereitstellt. Wenn dieser Pfad gefunden wird, wird Attach Implements Manually für dieses Fahrzeug deaktiviert.

Table 6: Parameter der `attachImplementsManuallyConfig.xml` (Teil 3)

1.6.3 Eigene Frontlader-AttacherJoint-Typen

Eigene AttacherJoint-Typen werden über `customFrontloaderJointTypes` eingetragen. Dadurch erkennt Attach Implements Manually Fahrzeuge mit diesen Joint-Typen als Frontlader-Fahrzeuge und kann die Frontlader-Kupplungstyp-Konfiguration verwenden.

```
1 <attachImplementsManuallyConfig>
2     <customFrontloaderJointTypes>
3         <customFrontloaderJointType name="MX" />
4     </customFrontloaderJointTypes>
5 </attachImplementsManuallyConfig>
```

Der eingetragene Name wird intern zu `JOINTTYPE_` plus Großbuchstaben erweitert. `name="MX"`

© 2022 - 2026 BayernGamers

entspricht also `JOINTTYPE_MX`. Der passende Joint-Typ kann anschließend in der Fahrzeug-XML wie gewohnt als `jointType` verwendet werden.

1.6.4 Fahrzeuge deaktivieren

Mit `disabledVehicles` können einzelne Fahrzeuge von Attach Implements Manually ausgeschlossen werden. Das kann sinnvoll sein, wenn ein Fahrzeug nicht kompatibel ist oder kein manuelles Ankuppeln unterstützen soll.

```
<attachImplementsManuallyConfig>
    <disabledVehicles>
        <disabledVehicle xmlFilename="vehicles/specialVehicle/specialVehicle.xml" />
    </disabledVehicles>
</attachImplementsManuallyConfig>
```

Für normale Modder-Konfigurationen reicht `xmlFilename`. Der Pfad wird relativ zum eigenen Mod aufgelöst. Das Attribut `modName` wird nur in der Konfigurationsdatei von Attach Implements Manually selbst ausgewertet und in anderen Mods ignoriert.

1.6.5 Eigene Schlauchgruppen

Schläuche mit einem Typ aus dem Basisspiel (`hydraulicIn`, `electric`, ...) werden automatisch ihrer Schlauchgruppe zugeordnet. Schläuche mit eigenem Typ werden in der Gruppe `custom` angeboten. Mit `customHoseGroups` kann ein eigener Schlauchtyp in einer anderen Gruppe angeboten werden, zum Beispiel über die Taste für Elektrik.

Das Attribut `provides` ist für Multikupplungen gedacht. Der Schlauch bleibt in der unter `group` angegebenen Gruppe, zählt im verbundenen Zustand aber zusätzlich als Verbindung der aufgeführten Gruppen. Ein Gerät mit einer Multikupplung, die `hydraulicIn hydraulicOut` bereitstellt, kann so zum Beispiel ohne separate Hydraulikschläuche hydraulisch bedient werden.

```
<attachImplementsManuallyConfig>
    <customHoseGroups>
        <customHoseGroup name="myElectricConnector" group="electric" />
        <customHoseGroup name="myMultiCoupler" group="custom" provides="hydraulicIn␣
            hydraulicOut␣electric␣isobus" />
    </customHoseGroups>
</attachImplementsManuallyConfig>
```

Unbekannte Gruppennamen werden ignoriert und als Warnung ins Log geschrieben.

© 2022 - 2026 BayernGamers

1.6.6 Automatisch gekuppelte eigene Schläuche

Eigene Schlauchtypen, die unter `autoCoupleCustomHoseTypes` eingetragen sind, werden nicht manuell gekuppelt. Sie werden zusammen mit dem Gerät automatisch verbunden und getrennt, zum Beispiel Datenkabel von Kamerasystemen.

```
<attachImplementsManuallyConfig>
    <autoCoupleCustomHoseTypes>
        <autoCoupleCustomHoseType name="myCamera*" />
    </autoCoupleCustomHoseTypes>
</attachImplementsManuallyConfig>
```

1.6.7 Handbremse überschreiben

Mit `handbrakeVehicles` lassen sich Fahrzeuge korrigieren, deren XML einen falschen Joint-Typ oder eine falsche Ausstattung angibt, sodass die automatische Erkennung die falsche Handbremse wählt. Ein Eintrag in der Konfigurationsdatei hat Vorrang vor der Fahrzeug-XML und vor der Shop-Konfiguration. Bis auf `xmlFilename` sind alle Attribute optional.

```
<attachImplementsManuallyConfig>
    <handbrakeVehicles>
        <handbrakeVehicle xmlFilename="vehicles/myTrailer/myTrailer.xml" type="
    SPRING_BRAKE" animationName="handbrakeLever" useWheelChocks="true" force="2" />
    </handbrakeVehicles>
</attachImplementsManuallyConfig>
```


© 2022 - 2026 BayernGamers

2 English

2.1 Adjusting Trigger Size for Vehicles

2.1.1 Introduction

This section describes how to adjust the trigger size for vehicles. This can sometimes be necessary if the vehicle size defined in the XML does not match the actual dimensions of the vehicle, or if the trigger should be placed at a specific position on the vehicle and, if required, be significantly smaller or larger than the vehicle itself. Please note that only one trigger is created per vehicle.

Starting with version 1.0.1.0, the trigger is no longer a physical trigger but a box-shaped detection area in which the player position is checked. This area is also used for the handbrake and the wheel chocks of trailers. The XML attributes are unchanged.

2.1.2 Adjusting the Trigger Size

By default, a vehicle's trigger size is determined by the vehicle dimensions defined in the XML. Ideally, these should cover the entire vehicle in height, width, and length. The vehicle dimensions can be visualized in-game using the console command `gsVehicleDebug`. After enabling it, a kind of menu bar appears at the bottom of the screen. For us, the `Attributes` entry is relevant, which can be selected via `Shift + 5`. After that, several boxes and text elements appear around the current vehicle. The dark blue box represents the vehicle's dimensions. Figure 2 shows the visualization of the vehicle dimensions using the console command `gsVehicleDebug`.

© 2022 - 2026 BayernGamers



Figure 2: Visualization of the vehicle dimensions using the console command `gsVehicleDebug`

The vehicle dimension values are defined in the vehicle XML via the attributes `length`, `width`, and `height`. In addition, an offset can be defined via the attributes `lengthOffset`,

© 2022 - 2026 BayernGamers

`widthOffset`, and `heightOffset`. These values shift the trigger relative to the vehicle's center.

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<vehicle type="tractor" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
    noNamespaceSchemaLocation="../../../../shared/xml/schema/vehicle.xsd">
    <annotation>Copyright (C) GIANTS Software GmbH, All Rights Reserved.</annotation
        >

    <storeData>
        [...]
    </storeData>

    <base>
        [...]
        <!-- Definition of the vehicle dimensions -->
        <size width="2.75" length="5.2" height="3.2" lengthOffset="0.35"/>
        [...]
    </base>
    [...]
</vehicle>
```

These vehicle dimensions then determine the size of the trigger that Attach Implements Manually uses for player detection. The vehicle's length is multiplied by a factor of 1.2, and the vehicle's width is multiplied by a factor of 1.3. The width factor is additionally increased to 2.3 if the vehicle's width is less than 2 meters. Starting with version 1.0.1.0, configurations that change the vehicle dimensions are taken into account as well, for example a different trailer length.

To change the trigger dimensions, the following entry can be added to the vehicle XML:

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<vehicle type="tractor" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
    noNamespaceSchemaLocation="../../../../shared/xml/schema/vehicle.xsd">
    <annotation>Copyright (C) GIANTS Software GmbH, All Rights Reserved.</annotation
        >

    <storeData>
        [...]
    </storeData>

    <base>
        [...]
    </base>
    [...]
    <attachImplementsManually>
        <trigger rootNode="0>1|0" offset="0␣1.5␣0" size="3␣3.5␣6"/>
    </attachImplementsManually>
</vehicle>
```


© 2022 - 2026 BayernGamers

The `rootNode` attribute defines which transform group the trigger is aligned to. If it is not set, the root node of the vehicle is used. The `offset` and `size` attributes define the trigger's position and size.

`size` defines the width, height and length of the trigger in meters. The values fully replace the size calculated from the vehicle dimensions, so the factors mentioned above are not applied. In the shown example, the trigger is 3 meters wide, 3.5 meters high and 6 meters long.

`offset` defines the position of the trigger's center relative to `rootNode` in meters (x, y, z). It overrides the offset values that Attach Implements Manually reads from the vehicle dimensions by default. In the shown example, the center of the trigger is 1.5 meters above `rootNode`.

Up to version 1.0.0.4, this guide described `size` as a scaling relative to the vehicle dimensions. This was wrong: the mod has always used the values as dimensions in meters. Existing entries therefore behave as before, but should be checked against the description above.

2.2 Adjusting Coupling Types for Front Loaders

2.2.1 Introduction

Attach Implements Manually automatically adds a new configuration option for front loaders, telescopic loaders, wheel loaders, and compact loaders, allowing the player to switch between three different coupling types. The player can choose between the coupling types `MANUAL`, `SEMI_AUTOMATIC`, and `AUTOMATIC`. To give modders the option to restrict the available coupling types for a vehicle, Attach Implements Manually allows defining the available coupling types in the vehicle XML. A default coupling type can also be defined, which is preselected when purchasing the vehicle.

© 2022 - 2026 BayernGamers

2.2.2 Adjusting the Coupling Types

Coupling type	Description	Use case
MANUAL	The player must attach the implement and connect the hoses manually.	For vehicles where, in reality, the implement lock must be engaged manually. Hoses must be connected manually.
SEMI_AUTO-MATIC	The player can couple an implement from inside the vehicle using `Q`. Hoses must be connected manually.	For vehicles that, in reality, have an automatically locking mechanism or a locking mechanism that can be operated from inside the cab. Hoses still have to be connected manually.
AUTOMATIC	The player can couple an implement from inside the vehicle using `Q`. Hoses are connected automatically.	For vehicles that, in reality, have an automatically locking mechanism or a locking mechanism that can be operated from inside the cab and are equipped with a hydraulic quick coupler.

Table 7: Overview of the different coupling types

By default, all coupling types are enabled so that the player can freely choose which coupling type to use. However, it is also possible to restrict the available coupling types for a vehicle. To do so, the following entry must be added to the vehicle XML:

```
1  <?xml version="1.0" encoding="utf-8" standalone="no"?>
2  <vehicle type="tractor" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:
       noNamespaceSchemaLocation="../../../../shared/xml/schema/vehicle.xsd">
3      <annotation>Copyright (C) GIANTS Software GmbH, All Rights Reserved.</annotation
           >
4
5      <storeData>
6          [...]
7      </storeData>
8
9      <base>
10         [...]
11     </base>
12     [...]
13     <attachImplementsManually>
```


© 2022 - 2026 BayernGamers

```
14          <defaultFrontloaderAttacherType type="SEMI_AUTOMATIC" allowSelection="true"
        allowedTypes="SEMI_AUTOMATIC␣AUTOMATIC"/>
15      </attachImplementsManually>
16  </vehicle>
```

The shown example allows the player to choose between the coupling types `SEMI_AUTOMATIC` and `AUTOMATIC`. The coupling type `MANUAL` is not available in this case. The `defaultFrontloaderAttacherType` attribute defines the coupling type that is selected by default when purchasing the vehicle. The `allowSelection` attribute defines whether the player should be able to choose between different coupling types. Finally, `allowedTypes` defines which coupling types should be available for selection. If `allowSelection` is set to `false`, the coupling type defined in `defaultFrontloaderAttacherType` is fixed for the vehicle and cannot be changed.

2.3 Input Attacher Joint Options

2.3.1 Introduction

Attach Implements Manually adds a few optional attributes to every `inputAttacherJoint` of an implement. They are read both in `vehicle.attachable.inputAttacherJoints` and in `inputAttacherJointConfigurations`. They allow an implement to couple automatically, or to be coupled from inside the cab, without changing the global settings.

2.3.2 Adjusting the Input Attacher Joint

```
1  <vehicle type="trailer">
2      [...]
3      <attachable>
4          <inputAttacherJoints>
5              <inputAttacherJoint node="attacherJoint" jointType="trailer" autoAttach=
       "true" autoAttachDistance="0.15" allowAttachFromInside="false"
       attachHosesIfAttachedFromInside="false" attachPTOIfAttachedFromInside="false"/>
6          </inputAttacherJoints>
7      </attachable>
8      [...]
9  </vehicle>
```


© 2022 - 2026 BayernGamers

Attribute	Type / Default	Description
`autoAttach`	Bool, `false`	The implement couples automatically as soon as the attacher joint of the vehicle is close enough, like trailers with a drawbar.
`autoAttachDistance`	Float, `0.1`	Distance in meters for `autoAttach`. Values below `0.001` are raised to `0.001`.
`allowAttachFromInside`	Bool, `false`	The implement can be coupled from inside the cab with `Q`, as without Attach Implements Manually.
`attachHosesIf AttachedFromInside`	Bool, `false`	The hoses are connected automatically when the implement is coupled from inside the cab.
`attachPTOIf AttachedFromInside`	Bool, `false`	The PTO shaft is connected automatically when the implement is coupled from inside the cab.

Table 8: Input attacher joint attributes of Attach Implements Manually

2.4 Hose Status of Hook Lifts

On a hook lift, the hoses to a hook lift container are only offered while the hook lift is unfolded. Some hook lift vehicles use an inverted folding state. For these vehicles, the hose status can be inverted in the XML of the hook lift vehicle:

```
<vehicle type="hookLiftTrailer">
    [...]
    <attachImplementsManually>
        <hookLift invertHoseStatus="true"/>
    </attachImplementsManually>
</vehicle>
```

2.5 Handbrake and Wheel Chocks

2.5.1 Introduction

Attach Implements Manually adds the shop configuration `handbrakeType` with the types `NONE`, `HANDBRAKE` and `SPRING_BRAKE`. The configuration is added automatically to every towed vehicle with wheels. A vehicle counts as towed if one of its input attacher joints uses one of the

© 2022 - 2026 BayernGamers

joint types `trailer`, `trailerLow`, `trailerCar`, `trailerSaddled`, `semitrailer`, `semitrailerHook`, `semitrailerCar`, `cutterTrailer` or `hookLift`. Sowing machines with an air system get the configuration as well.

Without an entry in the vehicle XML, the default type is detected automatically:

- `SPRING_BRAKE` if the vehicle has an air system (`attachable.airConsumer#usage` greater than 0, or an `airDoubleRed` / `airDoubleYellow` hose or target),
- otherwise `HANDBRAKE` if `attachable.brakeForce#force` is greater than 0,
- otherwise `NONE`.

`SPRING_BRAKE` can only be selected for vehicles with an air system, unless it is the default type.

2.5.2 Adjusting the Handbrake

```
<vehicle type="trailer">
    [...]
    <attachImplementsManually>
        <handbrake type="SPRING_BRAKE" allowSelection="true" allowedTypes="HANDBRAKE␣SPRING_BRAKE" animationName="handbrakeLever" useWheelChocks="true" force="2" pressureLossPerMinute="0.02"/>
    </attachImplementsManually>
</vehicle>
```


© 2022 - 2026 BayernGamers

Attribute	Type / Default	Description
`type`	String	Default type: `NONE`, `HANDBRAKE` or `SPRING_BRAKE`. If set, the vehicle gets the configuration even if it is not detected as a towed vehicle.
`allowSelection`	Bool, `true`	If `false`, only the default type is available in the shop.
`allowedTypes`	String list	Types the player can select. If empty, all available types can be selected.
`animationName`	String	Animation that is played when the handbrake is applied and played backwards when it is released.
`useWheelChocks`	Bool	Enables or disables the manual wheel chock action. If not set, the action is available as soon as the wheels of the vehicle have `wheelChock` entries.
`force`	Float, `2`	Minimum brake force while the handbrake or the wheel chocks hold the vehicle, in the unit of `attachable.brakeForce#force`. Used for vehicles without a brake force of their own.
`pressureLossPerMinute`	Float, `0.02`	Pressure in bar that the spring brake reservoir loses per minute while the red air line is not connected.

Table 9: Attributes of `attachImplementsManually.handbrake`

The spring brake reservoir is filled up to 8.5 bar while the red air line is connected. A spring brake can only be released at the implement with at least 5 bar left in the reservoir. Each manual release uses 1 bar.

2.5.3 Wheel Chocks

Wheel chocks are independent of the handbrake type. They use the base game `wheelChock` entries of the wheels, so no additional entry is needed for vehicles that already have wheel chocks. A vehicle with placed wheel chocks does not roll away.

© 2022 - 2026 BayernGamers

2.6 Global Configuration File for Attach Implements Manually

2.6.1 Introduction

In addition to the settings in the vehicle XML, a mod can provide a global configuration file for Attach Implements Manually. This file is intended for settings that should not be placed directly inside a single vehicle XML entry. It can be used to mark custom front loader attacher joint types as front loader couplings, to assign custom hose types to a hose group, to couple custom hose types automatically, to override the handbrake equipment of individual vehicles and to exclude individual vehicles from Attach Implements Manually.

The file can have any name. By default, `attachImplementsManuallyConfig.xml` is used as the file name. The file must be placed inside the mod and linked in the `modDesc.xml` using the `<attachImplementsManually>` tag:

```
1 <?xml version="1.0" encoding="utf-8" standalone="no"?>
2 <modDesc descVersion="109">
3     [...]
4     <attachImplementsManually configFilename="attachImplementsManuallyConfig.xml" />
5 </modDesc>
```

The path in `configFilename` is relative to the mod root directory. If the file is stored in a subfolder named `config`, for example, use `config/attachImplementsManuallyConfig.xml`.

2.6.2 Configuration File Structure

The configuration file uses `attachImplementsManuallyConfig` as its root element. All entries are optional:

```
1  <?xml version="1.0" encoding="utf-8" standalone="no"?>
2  <attachImplementsManuallyConfig>
3      <customFrontloaderJointTypes>
4          <customFrontloaderJointType name="MX" />
5      </customFrontloaderJointTypes>
6
7      <customHoseGroups>
8          <customHoseGroup name="myMultiCoupler" group="custom" provides="hydraulicIn␣
       hydraulicOut␣electric" />
9      </customHoseGroups>
10
11     <autoCoupleCustomHoseTypes>
12         <autoCoupleCustomHoseType name="myCamera*" />
13     </autoCoupleCustomHoseTypes>
14
15     <handbrakeVehicles>
16         <handbrakeVehicle xmlFilename="vehicles/myTrailer/myTrailer.xml" type="
       SPRING_BRAKE" />
```


© 2022 - 2026 BayernGamers

```
    </handbrakeVehicles>

    <disabledVehicles>
        <disabledVehicle xmlFilename="vehicles/myVehicle/myVehicle.xml" />
    </disabledVehicles>
</attachImplementsManuallyConfig>
```

Element / Attribute	Type	Description
`customFrontloader JointTypes`	Element	Container for custom attacher joint types that Attach Implements Manually should treat as front loader attacher joints.
`customFrontloaderJointType #name`	String	Name of the custom joint type without the `JOINTTYPE_` prefix. Example: For `name="MX"`, `JOINTTYPE_MX` is used internally. The joint type must already be registered as an attacher joint type in the game.

Table 10: Parameters of `attachImplementsManuallyConfig.xml` (part 1)

© 2022 - 2026 BayernGamers

Element / Attribute	Type	Description
`customHoseGroups`	Element	Container for custom hose types that are assigned to a hose group.
`customHoseGroup` `#name`	String	Custom hose type (`type` of a `customHose` / `customTarget`). Case-insensitive. A trailing `*` matches all types with this prefix.
`customHoseGroup` `#group`	String	Hose group the type is offered in: `hydraulicIn`, `hydraulicOut`, `airDoubleRed`, `airDoubleYellow`, `electric`, `isobus`, `custom` or `toolConnector`.
`customHoseGroup` `#provides`	String	Optional. Space-separated hose groups the type supplies in addition, for example for multi couplers.
`autoCoupleCustomHoseTypes`	Element	Container for custom hose types that are always connected automatically.
`autoCoupleCustomHoseType` `#name`	String	Custom hose type. Case-insensitive. A trailing `*` matches all types with this prefix.

Table 11: Parameters of `attachImplementsManuallyConfig.xml` (part 2)

© 2022 - 2026 BayernGamers

Element / Attribute	Type	Description
handbrakeVehicles	Element	Container for vehicles whose handbrake equipment is overridden.
handbrakeVehicle #xmlFilename	String	Path to the vehicle XML, relative to the root directory of the mod that provides the configuration file.
handbrakeVehicle #type	String	Optional. NONE, HANDBRAKE or SPRING_BRAKE.
handbrakeVehicle #animationName, #useWheelChocks, #force		Optional. Same meaning as in attachImplementsManually.handbrake of the vehicle XML.
disabledVehicles	Element	Container for vehicles where Attach Implements Manually should be disabled.
disabledVehicle #xmlFilename	String	Path to the vehicle XML, relative to the root directory of the mod that provides the configuration file. If this path is found, Attach Implements Manually is disabled for that vehicle.

Table 12: Parameters of attachImplementsManuallyConfig.xml (part 3)

2.6.3 Custom Front Loader Attacher Joint Types

Custom attacher joint types are defined via customFrontloaderJointTypes. This allows Attach Implements Manually to recognize vehicles using these joint types as front loader vehicles and to apply the front loader coupling type configuration.

```
1 <attachImplementsManuallyConfig>
2     <customFrontloaderJointTypes>
3         <customFrontloaderJointType name="MX" />
4     </customFrontloaderJointTypes>
5 </attachImplementsManuallyConfig>
```

The entered name is internally expanded to JOINTTYPE_ plus uppercase letters. Therefore, name="MX" corresponds to JOINTTYPE_MX. The matching joint type can then be used as usual as jointType in the vehicle XML.

© 2022 - 2026 BayernGamers

2.6.4 Disabling Vehicles

Use `disabledVehicles` to exclude individual vehicles from Attach Implements Manually. This can be useful if a vehicle is not compatible or should not support manual attaching.

```
<attachImplementsManuallyConfig>
    <disabledVehicles>
        <disabledVehicle xmlFilename="vehicles/specialVehicle/specialVehicle.xml" />
    </disabledVehicles>
</attachImplementsManuallyConfig>
```

For normal modder configurations, `xmlFilename` is sufficient. The path is resolved relative to your own mod. The attribute `modName` is only evaluated in the configuration file of Attach Implements Manually itself and is ignored in other mods.

2.6.5 Custom Hose Groups

Hoses with a base game type (`hydraulicIn`, `electric`, ...) are assigned to their hose group automatically. Hoses with a custom type are offered in the group `custom`. Use `customHoseGroups` to offer a custom hose type in another group, for example with the electric key.

The attribute `provides` is intended for multi couplers. The hose stays in the group given in `group`, but when it is connected it also counts as a connection of the listed groups. For example, an implement with a multi coupler that provides `hydraulicIn hydraulicOut` can be operated hydraulically without separate hydraulic hoses.

```
<attachImplementsManuallyConfig>
    <customHoseGroups>
        <customHoseGroup name="myElectricConnector" group="electric" />
        <customHoseGroup name="myMultiCoupler" group="custom" provides="hydraulicIn␣
            hydraulicOut␣electric␣isobus" />
    </customHoseGroups>
</attachImplementsManuallyConfig>
```

Unknown group names are ignored and logged as a warning.

2.6.6 Automatically Coupled Custom Hoses

Custom hose types listed in `autoCoupleCustomHoseTypes` are not coupled manually. They are connected and disconnected automatically together with the implement, for example data cables of camera systems.

```
<attachImplementsManuallyConfig>
    <autoCoupleCustomHoseTypes>
        <autoCoupleCustomHoseType name="myCamera*" />
    </autoCoupleCustomHoseTypes>
```


© 2022 - 2026 BayernGamers

```
</attachImplementsManuallyConfig>
```

2.6.7 Overriding the Handbrake

Use `handbrakeVehicles` for vehicles whose XML declares the wrong joint type or equipment, so that the automatic detection picks the wrong handbrake. An entry in the configuration file takes precedence over the vehicle XML and over the shop configuration. All attributes except `xmlFilename` are optional.

```
<attachImplementsManuallyConfig>
    <handbrakeVehicles>
        <handbrakeVehicle xmlFilename="vehicles/myTrailer/myTrailer.xml" type="SPRING_BRAKE" animationName="handbrakeLever" useWheelChocks="true" force="2" />
    </handbrakeVehicles>
</attachImplementsManuallyConfig>
```


© 2022 - 2026 BayernGamers