

Transformatoren für Fahrzeuge

Inhaltsverzeichnis

- [1 Transformatoren für Fahrzeuge in Transport Fever 3](#)
- [2 Welche Dateien gehören zu einem Transformator?](#)
 - [2.1 Die .trf.lua](#)
 - [2.2 Das Script](#)
 - [2.3 Hilfsdateien](#)
- [3 Einbindung in das Fahrzeug](#)
- [4 Events/Trigger im Fahrzeugmodell verwenden](#)
- [5 Praktische Beispiele](#)
 - [5.1 Tag- und Nachtsteuerung](#)
 - [5.2 Richtungs- und tageszeitabhängiger Zugschluss](#)
 - [5.3 Regenabhängige Bauteile](#)
 - [5.4 Scheibenwischer bei Regen](#)
 - [5.5 Geschwindigkeitsabhängige Events/Trigger](#)
 - [5.6 Geschwindigkeitsabhängiger Dampf und Rauch](#)
- [6 Erweiterter Train-Transformator MaikC/Skyjoe](#)
- [7 Hinweise und Fehlersuche](#)
- [8 Weiterführende Informationen](#)

1 Transformatoren für Fahrzeuge in Transport Fever 3

Mit Transport Fever 3 gibt es für Fahrzeuge eine neue Möglichkeit, während des laufenden Spiels auf verschiedene Zustände und Informationen zu reagieren: den **Transformator**.

Ein Transformator kann Informationen über das Fahrzeug und die Spielwelt auswerten und daraus **Events/Trigger** erzeugen, auf die das Fahrzeugmodell reagieren kann. Dadurch lassen sich Animationen und sichtbare Fahrzeugzustände deutlich flexibler steuern.

Mögliche Anwendungen sind zum Beispiel:

- Beleuchtung abhängig von Tag und Nacht
- Zugschluss abhängig von Fahrtrichtung und Tageszeit
- wetterabhängige Bauteile
- Scheibenwischer bei Regen
- geschwindigkeitsabhängige Animationen
- dynamisch angepasste Partikeleffekte

Die dabei verfügbaren Informationen beschränken sich nicht auf diese Beispiele. Dieser Artikel soll bewusst nur einen einfachen Einstieg in das Thema geben. Wer sich genauer mit den Möglichkeiten und den verfügbaren Daten beschäftigen möchte, findet diese in der offiziellen API-Dokumentation von Urban Games.

2 Welche Dateien gehören zu einem Transformator?

Ein Transformator besteht im einfachsten Fall aus einer `.trf.lua`-Datei und einem dazugehörigen Script.

Bei unserem Train-Transformator sieht das beispielsweise so aus:

Code

```
maikc_train_all.trf.lua  
maikc_train_all.script.tl  
maikc_transformator_util.tl (optional)
```

2.1 Die .trf.lua

Die .trf.lua ist die Verbindung zwischen dem Fahrzeug und den eigentlichen Scriptfunktionen. Sie legt fest, welche Funktionen des Transformators verwendet werden.

Neben der normalen Fahrzeugsteuerung können Transformatoren beispielsweise auch Partikeleffekte oder zusätzliche Modelle beeinflussen. Für einen einfachen Einstieg reicht es zunächst zu wissen, dass die .trf.lua auf die eigentliche Scriptlogik verweist.

2.2 Das Script

In der .script.tl befindet sich die eigentliche Logik. Dort werden Informationen aus dem Spiel ausgewertet und daraus Events/Trigger erzeugt.

Das kann beispielsweise vereinfacht so aussehen:

Code

```
Wenn                es                Nacht                Trigger ist:  
  
Wenn                es                regnet:  
                Trigger  
  
Wenn    das    Fahrzeug    schneller    als    10    km/h    fährt:  
    Trigger für hohe Geschwindigkeit einschalten
```

2.3 Hilfsdateien

Zusätzliche Hilfsdateien sind nicht erforderlich. Alle Funktionen könnten auch direkt in der eigentlichen .script.tl stehen.

Bei größeren Transformatoren kann es jedoch sinnvoll sein, wiederverwendbare Funktionen in eine separate Datei auszulagern. In unserem Beispiel verwenden wir dafür:

Code

```
maikc_transformator_util.tl
```

Das dient hauptsächlich der besseren Übersicht und Wartbarkeit.

Hilfsfunktionen können außerdem aus anderen Mods eingebunden werden, sofern diese dort entsprechend verfügbar sind. Dafür kann beispielsweise `ug_require` verwendet werden.

Code

```
local helper = ug_require "anderer_mod::/scripts/helper.tl"
```

Dadurch können bereits vorhandene Funktionen wiederverwendet werden, anstatt sie mehrfach in verschiedenen Mods zu pflegen.

Dabei entsteht allerdings eine Abhängigkeit zum anderen Mod. Ist dieser nicht geladen oder wird die eingebundene Datei geändert, kann auch der eigene Transformator nicht mehr korrekt funktionieren.

3 Einbindung in das Fahrzeug

Damit ein Fahrzeug einen Transformator verwendet, muss dieser in der .mdl des Fahrzeugs eingetragen werden.

Das geschieht über den Bereich transformatorConfig.

Code

```
transformatorConfig = {  
  
    },  
},
```

Damit weiß das Spiel, welcher Transformator für dieses Fahrzeug verwendet werden soll.

Ein Transformator kann von mehreren Fahrzeugen verwendet werden. Es ist also nicht notwendig, für jedes Fahrzeug eine eigene Transformator-Datei anzulegen.

Umgekehrt kann ein Fahrzeug jedoch nur **einen Transformator** verwenden. Mehrere verschiedene .trf.lua-Dateien können nicht gleichzeitig an dasselbe Fahrzeug gehängt werden.

Wer mehrere Funktionen benötigt, muss diese deshalb in einem gemeinsamen Transformator zusammenführen.

Auch zusätzliche Parameter können über transformatorConfig an den Transformator übergeben werden. Für einfache Anwendungen ist das zunächst aber nicht notwendig.

4 Events/Trigger im Fahrzeugmodell verwenden

Der Transformator bewegt nicht automatisch irgendwelche Bauteile des Fahrzeugs. Er erzeugt lediglich Events/Trigger.

Das Fahrzeugmodell muss anschließend festlegen, was bei einem bestimmten Trigger passieren soll.

Ein einfaches Beispiel wären:

Code

```
regen_on  
regen_off
```

Ein Bauteil im Fahrzeug kann auf diese Trigger reagieren. Bei regen_on könnte beispielsweise eine Dachluke geschlossen und bei regen_off wieder geöffnet werden.

Die entsprechenden Animationen werden wie andere Animationen direkt beim betreffenden Mesh im Modell eingetragen.

Code

```

animations
    ...
},
    ...
},
    ...
},
    ...
},
    ...
}

```

Nicht jedes Fahrzeug muss alle Events/Trigger eines Transformators verwenden.

5 Praktische Beispiele

5.1 Tag- und Nachtsteuerung

Unser Transformator kann die aktuelle Ingame-Tageszeit auswerten.

Dafür stehen folgende Events/Trigger zur Verfügung:

Code

```

parts_day_on
parts_day_off

parts_night_on
parts_night_off

```

Damit können beliebige Bauteile abhängig von der Tageszeit geschaltet werden.

Typische Anwendungen sind zum Beispiel:

- Innenbeleuchtung
- Führerstandsbeleuchtung



Externer Inhalt [youtu.be](https://youtu.be/...)

Inhalte von externen Seiten werden ohne Ihre Zustimmung nicht automatisch geladen und angezeigt.

[Alle externen Inhalte anzeigen](#)

Durch die Aktivierung der externen Inhalte erklären Sie sich damit einverstanden, dass personenbezogene Daten an Drittplattformen übermittelt werden. Mehr Informationen dazu haben wir in unserer Datenschutzerklärung zur Verfügung gestellt.

5.2 Richtungs- und tageszeitabhängiger Zugschluss

Transport Fever 3 kann bereits erkennen, an welcher Position sich ein Fahrzeug innerhalb eines Zuges befindet und in welche Richtung der Zug fährt.

Unser Transformator kombiniert diese Information zusätzlich mit der Ingame-Tageszeit.

Damit kann automatisch zwischen zwei verschiedenen Arten des Zugschlusses gewechselt werden:

- **Tagsüber:** Zugschluss tafeln
- **Nachts:** Zugschluss lampen

Die Tafeln beziehungsweise Lampen werden dabei nur an der Fahrzeugseite angezeigt, die tatsächlich das Zugende bildet.

Wird der Waggon gedreht, wechselt der Zugschluss automatisch auf die andere Seite.



5.3 Regenabhängige Bauteile

Der Transformator kann auch auf das aktuelle Wetter reagieren.

In unserem Beispiel werden dafür folgende Events/Trigger verwendet:

Code

```
regen_on  
regen_off
```

Als Grundlage wird die aktuelle Wolkenabdeckung verwendet. In meinen Tests begann sichtbarer Regen ungefähr bei einer Wolkenabdeckung von 90 Prozent.

Damit lassen sich beispielsweise:

- Dachluken schließen
- Fenster schließen
- Abdeckungen bewegen
- andere wetterabhängige Fahrzeugteile steuern

(Siehe Video unten)

5.4 Scheibenwischer bei Regen

Für Scheibenwischer reicht ein einfacher Regen-Trigger nicht aus. Bei Fahrzeugen mit zwei Führerständen muss zusätzlich berücksichtigt werden, welcher Führerstand aktuell vorne liegt.

Dafür verwendet unser Transformator:

Code

```
forward_rain  
backward_rain
```

Der Scheibenwischer wird dadurch nur auf der aktuell führenden Fahrzeugseite aktiviert.

Das bedeutet beispielsweise:

- bei Vorwärtsfahrt läuft nur der vordere Scheibenwischer,
- bei Rückwärtsfahrt läuft der Scheibenwischer des anderen Führerstands,
- bei einem Fahrzeug innerhalb des Zuges werden keine unnötigen Wischer aktiviert.

(Achtung das Video bitte manuell auf 1080p stellen, ich habe keine Ahnung warum es embedded nur in 360p abgespielt wird)

Externer Inhalt [youtu.be](https://www.youtube.com/watch?v=...)

Inhalte von externen Seiten werden ohne Ihre Zustimmung nicht automatisch geladen und angezeigt.

Alle externen Inhalte anzeigen

Durch die Aktivierung der externen Inhalte erklären Sie sich damit einverstanden, dass personenbezogene Daten an Drittplattformen übermittelt werden. Mehr Informationen dazu haben wir in unserer Datenschutzerklärung zur Verfügung gestellt.

5.5 Geschwindigkeitsabhängige Events/Trigger

Auch die aktuelle Geschwindigkeit eines Fahrzeugs kann ausgewertet werden.

Unser Transformator stellt dafür folgende Events/Trigger bereit:

Code

```
speed_low_on  
speed_low_off  
  
speed_high_on  
speed_high_off
```

Welche Funktion damit im Fahrzeug gesteuert wird, bleibt vollständig dem Fahrzeugmodder überlassen.

Mögliche Anwendungen wären beispielsweise:

- Lüfter
- Klappen
- bewegliche Fahrzeugteile
- Hypercar das bei 100km/h den Heckspoiler ausfährt
- andere geschwindigkeitsabhängige Animationen

5.6 Geschwindigkeitsabhängiger Dampf und Rauch

Transformatoren können nicht nur normale Fahrzeuganimationen steuern, sondern auch Partikeleffekte beeinflussen.

In unserem Train-Transformator werden bei bestimmten Dampf- und Raucheffekten Werte abhängig von der Fahrzeuggeschwindigkeit angepasst.

Dazu gehören unter anderem:

- Lebensdauer der Partikel
- Größe der Partikel

Zusätzlich wird bei den Partikeltypen der Farbversatz in schwarz bei Alterung auf null gesetzt.

Dadurch können Dampf- und Raucheffekte besser an das Verhalten des jeweiligen Fahrzeugs angepasst werden.

Externer Inhalt youtu.be

Inhalte von externen Seiten werden ohne Ihre Zustimmung nicht automatisch geladen und angezeigt.

Alle externen Inhalte anzeigen

Durch die Aktivierung der externen Inhalte erklären Sie sich damit einverstanden, dass personenbezogene Daten an Drittplattformen übermittelt werden. Mehr Informationen dazu haben wir in unserer Datenschutzerklärung zur Verfügung gestellt.

6 Erweiterter Train-Transformator MaikC/Skyjoe

Für unsere Fahrzeuge verwenden wir einen erweiterten Train-Transformator. Er basiert auf den normalen Funktionen des Train-Transformators und ergänzt diese um zusätzliche Events/Trigger und Anpassungen.

Aktuell enthalten sind:

- Tag- und Nachtsteuerung
- richtungs- und tageszeitabhängiger Zugschluss
- Regen-Trigger
- richtungsabhängige Scheibenwischer bei Regen
- geschwindigkeitsabhängige Events/Trigger
- geschwindigkeitsabhängige Anpassungen von Dampf- und Raucheffekten

Ein Fahrzeug muss dabei nicht alle Funktionen nutzen. Es werden einfach nur die Events/Trigger im Modell eingebaut, die für das jeweilige Fahrzeug benötigt werden.

7 Hinweise und Fehlersuche

Bei der Arbeit mit Transformatoren sollte man einige Dinge beachten.

- Ein Event/Trigger im Transformator bewirkt nur dann etwas, wenn das Fahrzeugmodell diesen auch verwendet.
- Ein Fahrzeug kann nur einen Transformator verwenden.
- Nicht alle Informationen stehen in jedem Fahrzeugkontext zur Verfügung.
- Vor dem Zugriff auf bestimmte Fahrzeugdaten sollte geprüft werden, ob diese vorhanden sind.
- Bei Fehlern in Transformator-Scripts sollte immer zuerst die `stdout.txt` geprüft werden.

Gerade bei eigenen Erweiterungen empfiehlt es sich, neue Funktionen zunächst mit einfachen Test-Events oder Debug-Ausgaben zu prüfen.

8 Weiterführende Informationen

Dieser Artikel soll nur einen praxisnahen Einstieg in Transformatoren geben und ersetzt nicht die vollständige technische Dokumentation.

Urban Games beschreibt im offiziellen Transport-Fever-3-Wiki unter anderem:

- die verschiedenen Transformator-Scriptfunktionen,
- die verfügbaren Fahrzeug- und Weltdaten,
- Animation States und Events,
- Partikelsysteme,
- das Emittieren zusätzlicher Modelle.

Für einen tieferen Einstieg empfiehlt sich deshalb die offizielle Dokumentation:

[Transport Fever 3 Wiki – Vehicle Advanced Topics](#)