

CommonAPI2 Bahnhofmodule mit eigenen Gleisen via Bahnhofsadapter

Inhaltsverzeichnis

- [1 Einführung](#)
- [2 Codebeispiel Gleis Modul](#)
 - [2.1 Erklärung](#)
 - [2.1.1 metadata Einträge:](#)
 - [2.1.2 updateFn](#)

1 Einführung

Build 29131

Da UG Ihren eigenen Weg für Gleismodule in den Sinn hat, bitte ich euch etwaige CommonAPI2 Gleismodule aus euren Schienenpaketen zu entfernen.

Daher ist dieses hier nur noch von historischer Bedeutung.

Zwar kann in TPF2 auch wieder Gleise erstellen, aber man hat wie unter TPF1 das Problem, das diese Gleise nicht einem Bahnhof genutzt werden können.
Damit es dann nicht X unterschiedliche Bahnhofsmodifikationen gibt

Hier hilft die CommonAPI2 + CommonAPI2 Bahnhofsadapter, damit ist es möglich mehrere Gleistypen zu unterstützen, darüber hinaus kann man damit auch Oberleitungen hinzufügen via Gleismodifikation.

Warum nicht selber einen Bahnhof editieren? Ja, das könnt Ihr natürlich, aber ihr werdet dann mit Supportfragen belästigt, warum Gleis X nicht mit Gleis Y funktioniert.

Wenn Ihr das untere Beispiel zeigt, wird keine feste Abhängigkeit zur CommonAPI2 geschaffen, Ihr könnt auch andere Lösungen anbieten. Auch andere Bahnhofkonstruktionen können diesen Standard folgen. Die CommonAPI2 hat noch ne ganze Menge an Hilfsfunktionen für die Bahnhofsanpassung.

2 Codebeispiel Gleis Modul

Code: gleis.module

```
metadata = {
    toggleCatenary = "station/rail/platform_eis_os_1000mm_track.module",
    toggleTrack = "station/rail/platform_eis_os_750mm_track_catenary.module",
},

updateFn = function(result, transform, tag, slotId, addModuleFn, params)
    if result.common_trackinterface.makeTrackByTrackType
    else
        -- Normal without CommonAPI2 (failback to normal track type)
        stationutil.makeTrack(result, transform, tag, slotId, addModuleFn, params)
    end
end,
```

Alles anzeigen

2.1 Erklärung

2.1.1 metadata Einträge:

In der Tabelle `common` müsst Ihr für das Gleismodifikationstool Informationen bereitstellen:

<code>hasCatenary</code>	<code>true/false</code>	Hat das Gleismodul eine Oberleitung (wichtig wenn man das Gleismodifikationstool nutzt) Zwingend Notwendig
<code>isHighSpeed</code>	<code>true/false</code>	Ob das Gleis eine Hochgeschwindigkeitsvariante darstellt. Zwingend Notwendig
<code>toggleCatenary</code>	<code>string</code>	Angabe des Modules wenn eine Oberleitung ab oder gebaut werden muss
<code>toggleTrack</code>	<code>string</code>	Angabe des Modules wenn der Gleistype umgeschaltet werden soll, von Normal auf Highspeed oder umgekehrt.

2.1.2 updateFn

Erst mal Abfragen ob der Bahnhof das CommonAPI2 Interface unterstützt:

```
if (result.common_trackinterface ~= nil) and
result.common_trackinterface.makeTrackByTrackType ~= nil) then
```

Dann können wir unseren Gleistyp bauen:

```
result.common_trackinterface.makeTrackByTrackType(result, transform, tag,  
slotId, addModuleFn, params, <Gleisname>, <Oberleitung>)
```

<Gleisname> string Euren Gleisdatei, ggf mit Unterverzeichnis Achtet auf **gross & KLEINSCHREIBUNG!**

<Oberleitung> true/false Mit Oberleitung bauen

else

Wenn der Bahnhoofsadapter und die CommonAPI2 nicht geladen wurden, oder dieser Bahnhof das Interface nicht unterstützt:

```
trainstationutil.makeTrack(result, transform, tag, slotId, addModuleFn, params,  
<ug gleistyp>)
```

Die UG Gleistypen <ug gleistyp> sind:

- Normal
- Normal mit Oberleitung
- Highspeed
- Highspeed mit Oberleitung

end