Bridge Configuration

Inhaltsverzeichnis

- 1 Bridge configuration
  - 1.1 materialsToReplace
  - 1.2 updateFn
- 2 Pfeiler
  - 2.1 configurePillar
- 3 Bridge girder

The information below is outdated as of June 22th 2020. Find the update description over at

https://transportfever2.com/wi…id=modding:bridgestunnels

[h1]

1 Bridge configuration

[/h1]

The bridge configurations are located in the folder `res/config/bridge`. They contain a `data()` function that returns the following data:

Name	Data type	Example value (iron.lua)	Description
name	String	_("Iron bridge")	The name of the bridge, which is shown as a tooltip in the bridge selection menu
yearFrom	Integer	1910	When will the bridge be available?
yearTo	Integer	0	Until when is the bridge available?
carriers	List from "RAIL", "ROAD"	{ "RAIL" , "ROAD"}	Is the bridge available as a road and/or railway bridge?
speedLimit	Double	180.0 / 3.6	Permissible maximum speed in m/s
pillarLen	Integer	3	(?) Length of a pier in bridge direction
pillarMinDist	Double	18.0	minimum distance between 2 pillars
pillarMaxDist	Double	66.0	maximum distance between 2 pillars

pillarTargetDist	Double	36.0	optimum distance between 2 pillars
cost	Double	300.0	Cost factor
materialsToReplace	List (see below)	...	Table for the replacement of textures for roads on bridges
updateFn	Function (see below)	bridgeutil.makeDefaultUpdateFn(config)	Function used for the composition of the bridge models

1.1 materialsToReplace

This table is used to exchange materials for roads when they are built on bridges. All materials that are also specified for roads are possible:

- streetPaving, streetBorder, streetLane, streetStripe, streetStripeMedian
- streetTram, streetTramTrack, streetBus
- crossingLane, crossingBus, crossingTram, crossingTramTrack, crossingCrosswalk, crossingStopline
- sidewalkPaving, sidewalkLane, sidewalkBorderInner, sidewalkBorderOuter, sidewalkCurb, sidewalkWall

1.2 updateFn

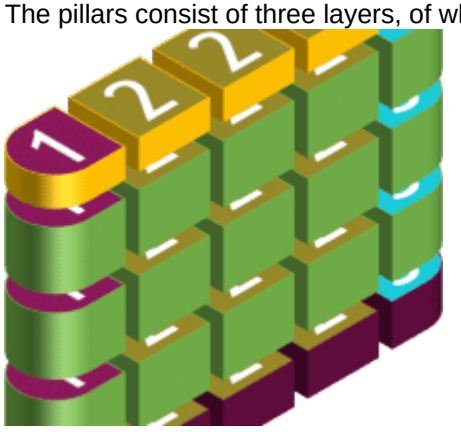
This function is used to construct the bridge model from individual parts. In `scripts/bridgeutil.lua` a prefabricated function is offered, which is also used by the vanilla bridges. This function gets a config table as parameter. The contents of this table are described below:

[h1]

2 Pfeiler

[/h1]

The pillars consist of three layers, of which the middle layer can be repeated several times:



pillarBase (purple)
Dies ist eine Liste mit Modellen, die für den unteren Abschluss eines Pfeilers genutzt werden. Die Länge der Liste kann variieren, möglich sind:

1 Model		This model is set in the middle as a pillar.
2 Models		The first model is used for the edges of the pier (rotated once accordingly), the second model is placed next to each other for the middle of the pier and scaled slightly until it fills the required width.
3 Models		The first model is used for one side of the pillar, the second model is placed next to each other for the middle of the pillar and scaled slightly until it fills the required width and the third model is used for the other side, but not rotated.

pillarRepeat (green)
This is a list of models used for the height variable part of the pillar. These models are strung together vertically several times and, if necessary, scaled slightly to reach the required height. The length of the list can vary, possible combinations are as described for **pillarBase** .

pillarTop (yellow)
This is a list of models used for the upper end of the pillar. The length of the list can vary, possible combinations are as described for **pillarBase** .

2.1 configurePillar

If you do not want to use the function described above to assemble the columns, but instead assemble another column based on the height of the column, for example, you can specify a separate function for this:

Lua

```
configurePillar = function(modelData, params, i, height, width) [...] end,
```

This has the following parameters:

Name	Data type	Description
modelData	complex list	A list of bridge element models and their properties For further information see below

params	complex list	A list with the parameters of the bridge section currently under construction (mostly up to 3 piers long). For further information see below
i	Integer	The number of the pillar in the bridge section currently under construction
height	Double	The height of the pillar to be built in meters
width	Double	The width of the pillar to be built in meters

modelData
The default configurePillar function passes a list of all loaded bridge element models (not only those of this bridge!). The path to the mdl file is used as the key of the list entries. For each entry the BoundingBox is provided:

Lua

```
["bridge/iron/pillar_top_side.mdl"]                    =                    {
```

```
} ,
```

Alles anzeigen

Due to the game loading and extracting all these Bounding Box data, it is required that all bridge related mdl files are located under `res/models/model/bridge/`!

params
At runtime, a table with various parameters is delivered with the method call. These are:

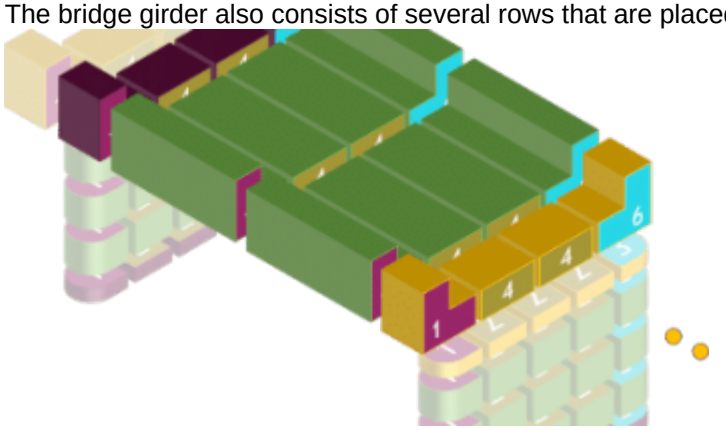
- `pillarWidth`: The width of a pillar
- `pillarHeights`: A list with the heights of the pillars
- `pillarLength`: The length of a pillar
- `state.models`: A list with all loaded bridge models (see above)
- `railingWidth`: The width of the bridge
- `railingIntervals`: A list of bridge sections. There is three pieces of information for each section:
    - `hasPillar`: A pair of two boolean values that probably tell whether a pillar is placed at the beginning and end of the section
    - `lanes`: ?
    - `length`: The length of the section

[h1]

3 Bridge girder

[/h1]

The bridge girder also consists of several rows that are placed next to each other:



railingBegin

This is a list of models that are used for the beginning of a bridge girder segment. A bridge segment starts at the beginning of a bridge and at each pier. There can be either 5 or 8 models in the list. Unfortunately for some elements of the list it is not yet clear what they mean. If only 5 elements are included, the elements 1-3 are used as a substitute for 6-8 in a rotated version:

5 Models

The first model is used for the outer sides (rotated once accordingly), the fourth model is placed next to each other for the middle of the line and scaled slightly until it fills the required width.

8 Models

The first model is used for one outside, the fourth model is placed next to each other for the middle of the line and scaled slightly until it fills the required width and the sixth model is used for the other outside but does not rotate.

railingRepeat

This is a list of models that are used for the center of a bridge girder segment. These models are lined up

horizontally several times and, if necessary, scaled slightly to reach the required height. The length of the list can vary, possible combinations are as described for **railingBegin** .

railingEnd
This is a list of models that are used for the end of a bridge girder segment. The length of the list can vary, possible combinations are as described for **railingBegin** .