

# Python Gleisbildeditor

Ergänzung zum Beitrag in der DiMo Heft 3-2024: Die Schlange im Stellwerk, Teil 2

## Copyright-Hinweis:

```
# Lizenzbestimmungen:  
# Autor: Friedrich Bollow  
# Lizenzvertrag: CC BY-NC-SA 4.0  
# Kurzform: Namensnennung - Nicht-kommerziell - Weitergabe unter  
# gleichen Bedingungen 4.0 International  
# Details: creativecommons.org/licenses/by-nc-sa/4.0/legalcode.de
```

## Size

Beim ersten Starten ist eine default-Einstellung der Bildschirmgröße vorhanden:  
Horizontal 15 Gleisbildelemente, Vertikal 16 Gleisbildelemente.  
Diese Größe passt z.B. zur Anlage pySTW19.csv.

Die Bildschirmgröße ist über das Menü "size" anpassbar. Die Bildschirmgröße kann bei einem geladenen Gleisplan verändert werden. Dies gilt auch dann, wenn der Gleisplan nicht auf die vorhandene, sichtbare Gleisbildgröße geladen wird.

Eine geänderte Größe ist in der Datei Panelsize.txt gespeichert. Bei Folgeaufrufen des Gleisbildeditors erfolgt der Start mit der geänderten Größe.

## move und copy

Der neue Platz sollte die verschobenen bzw. neuen Elemente aufnehmen können.  
Belegte Flächen auf dem Gleisbild werden überschrieben.  
Vor dem Verschieben das Gleisbild sichern. Ggf. zum Verschieben eine größere Fläche einstellen. Bei dem Verschieben oder Kopieren wird nur bis zur maximalen Größe verschoben. Darüber hinausweisende Elemente werden abgeschnitten.

Bei den Menüaufrufen von move/clear/copy entsteht die Datei opmeT.csv, die mit Load ebenfalls ladbar ist. Bei Bedarf sollte diese Datei mit anderem Namen gespeichert werden.

## **Gleissperre**

Die Gleissperre ist im Editor als normale Weiche (Typen 16,17,18,19) sichtbar.  
Die Betriebssoftware stellt in der nächsten Version die Gleissperre unterschiedlich zu den Weichen dar.

## **Links**

Links sind für die Betriebssoftware interessant, der Editor speichert lediglich Zeichenketten. Bei dem Verschieben muss das Ziel nachkorrigiert werden, künftig ist das Format `&t;11;7;:B;` vorgesehen. Dabei erhalten Ziel und Quelle den gleichen Großbuchstaben. Auf dem Gleisbild sind dann Quelle und Ziel sichtbar.

## **Betriebssoftware**

Die vom Gleisbildeditor generierten CSV-Dateien stellen den Input für die Betriebssoftware dar. Teil 1 der Beitragsreihe (DiMo Heft 1-2024) enthält eine erste Version, die noch nicht alle Funktionen der vollständigen Betriebssoftware enthält.  
Teil 3 der Beitragsreihe beschreibt die vollständige Betriebssoftware.

## **Korrekturen zum Beitrag DiMo Heft 3-2024: Die Schlange im Stellwerk, Teil 2**

Das Import-Modul „pygame“ ist nicht mehr Teil der Software.

Seite 68, 2. Absatz, 2. Satz:  
Richtig ist: „mit der linken Maustaste“.

## Python-Versionen

Die Entwicklung erfolgte auf Windows-PCs und auf Raspberry-Rechnern. Die Versionen der Betriebssysteme und der Module sind hier dargestellt:

Windows	Raspberry
PC 1: Dell	Raspberry 400
Windows-Version, Windows-Logo-Taste + R, Eingabe winver, ok → Windows 10, Version 22H2	
Version Python, Windows Command Prompt öffnen, Eingabe: python -version → C:\>python3 -V → Python 3.11.9	Console: python3 -V Python 3.9.2
C:\>python3 -m tkinter Mini Fenster: Tcl/Tk 8.6.12	python3 -m tkinter Mini Fenster: Tcl/Tk 8.6
PC 2: HP Windows 10, Version 22H2	Raspberry PI B+
C:\>python3 -V Python 3.11.9	Console python3 -V Python 2.8.16
C:\>python3 -m tkinter Mini Fenster: Tcl/Tk 8.6.12	python3 -m tkinter Mini Fenster: Tcl/Tk 8.6

## Apple Mac Computer

Bisher wurde lediglich die Python-Installation erfolgreich geprüft und die IDLE-Shell aufgerufen.

Der Autor betreibt eine Spur Null Anlage mit der z21 der Firma Roco (XBus).  
Test sind erfolgt mit LocoNet und der Intellibox sowie mit der Daisy II-Zentrale.

Bei Fragen oder Problemen zum Editor verwenden Sie bitte die beim Autor eingesetzte CSV-Datei pySTW12.csv. Dort sind auch alle Gleis-, Weichen und Signalkomponenten im Einsatz.

Kontakt zum Autor	XLshield@gmx.de
-------------------	-----------------