

# CHAPTER 3 - CONFIG AND KIND

## CONFIG.TXT

Each item of content that you create is required to have a config.txt file. This file is a simple text file that is used to describe the item of content to Trainz. The contents of a typical config file may look like this:

Blue text indicates Required tags

Green text indicates Optional tags.

```
kuid <KUID:1234:5678>
kuid-table {
    0 <KUID:1234:6000>
    1 <KUID:1234:6001>
}
obsolete-table {
    0 <KUID:1234:5676>
    1 <KUID:1234:5677>
}
username My Locomotive
description "This is an example Asset.
Note that you can have multiple lines but
no double quote characters in here. Trainz
automatically wraps this text."
region Australia
trainz-build 1.5
kind engine
category-class AD
category-region-0 US
category-region-1 UK
category-region-2 AU
category-era-0 1960s
category-era-1 1970s
author Greg Lane
organisation Trainz Thingz
contact-email helpdesk@auran.com
contact-website http://www.auran.com
```

*Side-note:  
(kuid-table must be  
included where the  
config references  
additional KUIDs,  
such as a bogey, or  
a pantograph. The  
Download Station  
performs a search, and  
those found are added  
to the download pack)*

## Breakdown of Config.txt

### kuid

Unique ID of this asset. Any subsequent versions of this asset MUST have a different KUID. The KUID contains basic creator information.

### kuid-table

A list of KUIDs required for this asset to function correctly.

### obsolete-table

A history of this asset. Each KUID listed is a previous version of this asset. If there are no previous versions display as follows:

```
obsolete-table {
}
```

### username

The human-readable English name of this asset.

### description

The human-readable multi-line English description of this asset.

### region

The country region to which this asset belongs. This should be one of the Auran-supplied region names

### trainz-build

The Trainz build number for which this asset was created.

### kind

The asset kind. Must be one of the Auran-supplied asset kinds.

### category-class

The class code for this asset. Classes are unique per kind of asset but may not be across all assets

### category-region-0

A list of REGION codes or REGION GROUP codes, starting at category-region-0 and progressing category-region-1, etc. with one code per tag

### category-era-0

A list of ERA codes, starting at category-era-0 and progressing category-era-1, etc. with one code per tag.

**KIND (Config variables)****KIND: ENGINE**

File location - \Trainz\Engines\

Locomotive performance parameters...

Config.txt :

```

kuid <KUID:-1:42004202>
kind engine
rem F7 Generic Engine
flowsize {
trainbrakepipe 170000
epreservoirpipe 0.1
no3pipe 0.1
no4pipe 0.1
auxreservoirvent 0.1
auxreservoir_no3 0.1
auxreservoir_trainbrakepipe 0.1
autobrakecylindervent 0.1
auxreservoir_autobrakecylinder 0.1
equaliser_mainreservoir 0.06
equaliservent 0.06
equaliserventhandleoff 0.1
equaliserventemergency 0.1
no3pipevent 1.5
no3pipe_mainreservoir 0.1
compressor 10
trainbrakepipe_reservoir 1
trainbrakepipevent 0.06
no3pipe_autobrakecylinder 0.1
epreservoirpipe_autobrakecylinder 0.1
mainreservoir_ep 0.1
vacuumbrakepipe 0.1
vacuumbrakepipereleasevent 0.1
vacuumbrakepipevent 0.1
vacuumbrakereservoir_vacuumbrakepipe
0.1
vacuumbrakecylinder_vacuumbrakepipe
0.1
highspeedexhauster_vacuumbrakepipe 0.1
}
volume {
scale 1
trainbrakepipe 0.2
epreservoirpipe 0.2
no3pipe 0.2
no4pipe 0.2
auxreservoir 0.0384678
autobrakecylinder 0.00969387
vacuumbrakepipe 0
vacuumbrakereservoir 0
vacuumbrakecylinder 0
mainreservoir 0.9
equaliser 0.5
independantbrakecylinder 0.0103239

```

```

}
pressure

{
scale 1
compressor
0.00946941
mainreservoir
0.00946941
highspeedexhauster 0
brakepipe 0.00665741
brakeinitial 0.00609501
brakefull 0.00504051
indbrakefull 0.00504051
trainbrakepipe_start 0.00504051
epreservoirpipe_start 0
no3pipe_start 0
no4pipe_start 0
auxreservoir_start 0.00504051
autobrakecylinder_start 0.00507566
vacuumbrakepipe_start 0
vacuumbrakereservoir_start 0
vacuumbrakecylinder_start 0
mainreservoir_start 0.00806341
equaliser_start 0.00504051
independantbrakecylinder_start
0.00507566
}
mass {
scale 1
fuel 6.2156e+006
}
motor {
resistance 1.7
adhesion 2.5
maxvoltage 600
maxspeed 40
brakeratio 55000
max-accel
3500
max-decel
9000
axle-count 4
surface-area 80
moving-friction-coefficient .03
air-drag-coefficient .00017
}

throttle-power
{
0 {
0 0
}
1 {
0 30
5 25
}

```

```

    10    15
    12    0
  }
2 {
    0    55
    5    48
    10   40
    15   30
    30   10
  }
3 {
    0    100
    5    70
    10   55
    15   50
    30   10
  }
4 {
    2    150
    5    80
    10   60
    15   55
    30   10
  }
5 {
    0    200
    5    150
    10   80
    15   65
    30   10
  }
6 {
    0    231
    5    180
    10   120
    15   90
    30   20
  }
7 {
    0    260
    5    220
    10   200
    15   110
    30   20
  }
8 {
    0    300
    3.5  250
    5    190
    10   160

```

```

    20    120
    30    110
    35    140
    44    20
  }
}

dynamic-brake
{
    0 {
        0    0
    }
    1 {
        1.333 0
        2    30
        5    25
        10   15
        12    0
    }
    2 {
        1.333 0
        3    50
        10   35
        14   20
        15    0
    }
    3 {
        1.333 0
        3    60
        10   40
        17   20
        22    0
    }
    4 {
        1.333 0
        4    80
        10   60
        20   20
        25    0
    }
    5 {
        1.333 0
        5    90
        10   70
        25   25
        29    0
    }
    6 {
        1.333 0
        5    150

```

```

        10      80
        29      70
        32      0
    }
    7 {
        1.333  0
        5      200
        10     100
        32     60
        36     0
    }
    8 {
        1.33  0
        5     200
        10    150
        36    50
        40    0
    }
}

region Australia
kuid-table {
    0 <KUID:###:#####>
    1 <KUID:###:#####>
}
obsolete-table {
}
username " "
description " "
trainz-build 1.5
category-class
category-region-0
category-era-0

```

## Breakdown of engine

*kind* – asset type

*rem* – comment used to display engine name

### flowsize

rate of flow through pipes, generally leave these settings:

```

trainbrakepipe 170000
epreservoirpipe 0.1
no3pipe 0.1
no4pipe 0.1
auxreservoirvent 0.1
auxreservoir_no3 0.1
auxreservoir_trainbrakepipe 0.1
autobrakecylindervent 0.1
auxreservoir_autobrakecylinder 0.1
equaliser_mainreservoir 0.06
equaliservent 0.06

```

```

equaliserventhandleoff 0.1
equaliserventemergency 0.1
no3pipevent 1.5
no3pipe_mainreservoir 0.1
compressor 10
trainbrakepipe_reservoir 1
trainbrakepipevent 0.06
no3pipe_autobrakecylinder 0.1
epreservoirpipe_autobrakecylinder 0.1
mainreservoir_ep 0.1
vacuumbrakepipe 0.1
vacuumbrakepipereleaseevent 0.1
vacuumbrakepipevent 0.1
vacuumbrakereservoir_vacuumbrakepipe 0.1
vacuumbrakecylinder_vacuumbrakepipe 0.1
highspeedexhauster_vacuumbrakepipe 0.1

```

**volume** – size of pipes and appliances

*scale* 1

*trainbrakepipe* ..... 0.2  
brake pipe volume

*epreservoirpipe* .....0.2  
For electro pneumatic braking - not currently in use, generally leave this setting

*no3pipe* ..... 0.2  
Independent brake pipe

*no4pipe* .....0.2  
Bail pipe - not currently in use, generally leave this setting

*Auxreservoir* .....0.0384678  
Auxiliary reservoir volume.

*Autobrakecylinder* ..... 0.00969387  
Brake cylinder volume.

*vacuumbrakepipe* .....0  
*vacuumbrakereservoir* ..... 0  
*vacuumbrakecylinder* ..... 0  
For vacuum braking - not currently in use, generally leave this setting

*mainreservoir* .....0.9  
Main reservoir volume.

*equaliser* .....0.5  
Equalising reservoir volume

*independantbrakecylinder* .....0.0103239  
Loco brake cylinder volume

**pressure**  
brake system pressures.

**scale 1**

multiplies pressure by given value, generally leave this setting.

**Compressor**.....0.00946941  
(120psi expressed in grams/m<sup>3</sup>)  
compressor maximum pressure.

**mainreservoir**.....0.00946941  
main reservoir maximum pressure

**highspeedexhauster** .....0  
For vacuum braking - not currently in use,  
generally leave this setting

**brakepipe** .....0.00665741  
(80psi expressed in grams/m<sup>3</sup>)  
brake pipe pressure when fully charged

**brakeinitial**.....0.00609501  
(72psi expressed in grams/m<sup>3</sup>)  
brake pipe pressure after initial service  
reduction (for self lapping brakes)

**brakefull** .....0.00504051  
(57psi expressed in grams/m<sup>3</sup>)  
Brake pipe pressure after full service reduction  
(for self lapping brakes)

**indbrakefull** .....0.00504051  
Brake cylinder pressure for independant brake  
service.

**trainbrakepipe\_start** .....0.00504051  
Brake pipe pressure on loading the game.

**epreservoirpipe\_start** .....0  
For electro pneumatic braking - not currently in  
use, generally leave this setting

**no3pipe\_start** .....0  
**no4pipe\_start** .....0  
Generally leave these settings.

**auxreservoir\_start**.....0.00504051  
Auxiliary reservoir pressure on loading the  
game.

**autobrakecylinder\_start** .....0.00504051  
Train brake cylinder pressure on loading the  
game.

**vacuumbrakepipe\_start** .....0  
**vacuumbrakereservoir\_start** .....0

**vacuumbrakecylinder\_start** .....0  
For vacuum braking - not currently in use,  
generally leave this setting

**mainreservoir\_start** .....0.00806341  
(100psi expressed in grams/m<sup>3</sup>)  
Main Reservoir pressure on loading the game.

**equaliser\_start** .....0.00504051  
Equalising Reservoir pressure on loading the  
game.

**independantbrakecylinder\_start** .....0.00504051  
Locomotive brake cylinder pressure on loading  
the game.

**mass****scale 1**

multiplies fuel mass by given value, not  
currently in use, generally leave this setting.

**fuel** .....6.2156e+006  
fuel level, not currently in use, generally leave  
this setting.

**motor**

**resistance**.....1.7  
power figure for DCC, higher resistance  
value=less power

**adhesion** .....2.5  
adhesion parameter, higher value=greater  
adhesion

**maxvoltage** .....600  
generally leave this setting

**maxspeed** .....40  
maximum speed for DCC, expressed in  
metres per second.

**Brakeratio**..... 55000  
brake force for pressure reduction

**max-accel**.....3500  
**max-decel**.....9000  
parameters for DCC acceleration &  
deceleration.

**axle-count**.....4  
Resistance – axle count

**surface-area** .....80  
Resistance – surface area

*moving-friction-coefficient*.....0.03

Resistance – moving friction

*air-drag-coefficient*.....0.00017

Resistance – air drag

### ***throttle-power***

*Acceleration variables in cabin mode*

```
1 {      = notch number (1)
    0      30
    5      25      = At speed 5
    10     15      acceleration = 25
    12     0
}
```

### ***dynamic-brake***

*Deceleration variables while dynamic braking in cabin mode*

```
1 {      = notch number (1)
    1.333  0
    2      30
    5      25      = At speed 5
    10     15      deceleration = 25
    12     0
}
```

## **Equalisation of Pressures**

There is a point at which no further brake pipe pressure reduction will result in increased braking effort, this is known as full application or equalisation of pressures.

Imagine you made a 26 psi reduction when operating a loco with a 90psi brake pipe. 90psi in the train pipe minus 26psi reduction equals 64 psi in the pipe. Due to the 2.5:1 ratio of auxiliary reservoir volume to brake cylinder volume, the 26 psi reduction puts 64 psi into the brake cylinder.

As the pressure in the reservoir and the pressure in the cylinder is now equal, no more air will flow into the brake cylinder; and making a further reduction in brake pipe pressure will have no effect on braking.

Equalisation occurs at different pressures, depending on the train pipe feed pressure.

100 psi pipe (e.g. the UK locos - 7 bar) equalisation at 71 psi.

90 psi pipe (e.g. the US locos) equalisation at 64 psi.

72 psi pipe (e.g. French & Queensland locos) equalisation at 49 psi.

The easiest way to set your custom content to the desired brake pipe feed pressure is to copy the entire **pressure** section from the config of a loco that uses the pressure you desire.

\*Note: Converting PSI to Grams /m cubed...

e.g. 90psi... (90+14.7).0000703

104.7 x .0000703=.00736041

**KIND: BOGEY**

File location - \World\custom\bogeys\

This is a bogey.

Referenced by the *bogey* tag in a traincar config.txt

Config.txt:

```
kuid <KUID:###:#####>
kind bogey
obsolete-table {
}
username
description
trainz-build 1.5
category-class AC
category-region-0 AT
category-era-0 1980s
animdist 2.1
```

**Breakdown:****animdist**

Leave this tag out if the bogey is not animated.

The distance travelled in meters by the bogeys in 1 second (30 frames) of animation. Bogey animations (exported from Gmax or 3ds Max) are called “anim.kin”.

Refer to Chapter 4 - Bogeys for modeling guidelines

**KIND: TRAINCAR**

File location - \World\custom\trains\

This is a locomotive or car item.

Config.txt:

```
kuid <KUID:###:#####>
kind traincar
origin AUT
name OBB 1044
company OBB
mass 84000
bogey <KUID:###:#####>
pantograph <KUID:###:#####>
interior <KUID:###:#####>
engine 1
fonts 2
running-numbers {
  rn-0 #104420
  rn-1 #104427
  rn-2 #104430
  rn-3 #104452
}
enginespec <KUID:-1:42004207>
enginesound <KUID:-1:42003002>
hornsound <KUID:-1:42003101>
smoke0
{
  attachment      a.steam.l
  mode             anim
  color            255,255,255,150

  start           0
  period          0.4
  rate            2
  velocity        1
  lifetime        2
  minsize         0.05
  maxsize         1
}
description " "
kuid-table {
  0 <KUID:###:#####>
  1 <KUID:###:#####>
  2 <KUID:###:#####>
}
obsolete-table {
}
username My locomotive
trainz-build 1.5
category-class AC
category-region-0 AT
category-era-0 1980s
light_color 255,255,255
ditch_color 255,200,200
```



**Breakdown of Traincar Config.txt:****origin**

The Country Abbreviation

**company**

The Locomotive or car owner

**mass**

Mass in kilograms

**bogey**

The bogey kuid number (default for a.bog0 and a.bog1)

**bogey-1**

The bogey kuid number for a.bog1 (Used only if different to a.bog0)

**bogey-r and bogey-1-r**

Used instead of 'bogey' and bogey-1. The bogey will have reversed orientation. Note: This will cause bogey animation to play in reverse.

**pantograph**

The pantograph kuid number inserted at a.pant0, a.pant1, etc. Use this tag only when needed.

**interior**

Kuid number of the required interior. Inserted at a.cabfront. Use this tag only when needed eg Locomotive's.

**engine**

States type of traincar.

**0** = Car

**1** = Locomotive

**fonts**

Indicates how many types of numbering fonts used.

E.g. **0** = no fonts used

**1** = one font

Digit textures (*digit\_1.tga to digit\_6.tga*) replaced automatically with alphanumeric textures (*alphanumeric\_0 to alphanumeric\_9*) as numbers are changed in 'My Collection'.

**2** = two fonts

Digit textures (*digit\_1a.tga to digit\_6a.tga and digit\_1b.tga to digit\_6b.tga*) replaced automatically with alphanumeric textures (*alphanumeric\_0a to alphanumeric\_9a and alphanumeric\_0b to alphanumeric\_9b*) as numbers are changed in 'My Collection'.

**running-numbers**

The default numbers viewed before being changed in 'My Collection'. Generally Auran uses 4 variations of numbers (up to 6 digits each). Note: This tag not required if *fonts 0* is used.

**enginespec**

The engine kuid number. This specifies the driver physics boundaries for the traincar.

Located in \Trainz\Engines\

Refer Chapter 3, KIND: TRAINCAR

**WARNING:**

**ALTERING *ENGINESPEC* FIGURES MAY RESULT IN UNDESIRE EFFECTS IN PERFORMNACE AND BEHAVIOR OF YOUR TRAINS. (MAKE BACK-UP COPIES OF YOUR ENGINE CONFIG FILES!!)**

**enginesound**

The kuid number for the traincar's sound.

Located in \world\custom\enginesound\

Refer Chapter 3, KIND: ENGINESOUND

**hornsound**

The kuid number for the traincar horn sound.

Located in \world\custom\hornsound\

Refer Chapter 3, KIND: HORNSOUND

**smoke0**

Sets boundaries for smoke, steam, vapor and similar effects. Refer: [Smoke Effects](#)

**description** “ “

Description of model for 'My Collection' information

**light\_color**

RGB headlight colour. Eg. 255,255,255

**ditch\_color**

RGB ditch light colour. Eg. 255,255,255

**KIND: ENGINESOUND**

File location - \World\custom\enginesound\

This is the traincar sound.

Referenced by the *enginesound* tag in a traincar config.txt

Config.txt:

```
kuid <KUID:###:#####>
kind enginesound
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class AC
category-region-0 AT
category-era-0 1980s
```

Sound .wav files to be located in the same subfolder as the enginesound config.txt.

down 2 - 1.wav	up 1 - 2.wav	idle 1.wav	start 1-1.wav
down 3 - 2.wav	up 2 - 3.wav	idle 2.wav	stop 1-1.wav
down 4 - 3.wav	up 3 - 4.wav	idle 3.wav	
down 5 - 4.wav	up 4 - 5.wav	idle 4.wav	
down 6 - 5.wav	up 5 - 6.wav	idle 5.wav	
down 7 - 6.wav	up 6 - 7.wav	idle 6.wav	
down 8 - 7.wav	up 7 - 8.wav	idle 7.wav	
		idle 8.wav	

**KIND: HORNSOUND**

File location - \World\custom\*hornsound*

This is the traincar horn sound.

Referenced by the *hornsound* tag in a traincar config.txt

Config.txt:

```
kuid <KUID:###:#####>
kind hornsound
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class
category-region-0
category-era-0
```

Sound .wav files to be located in the same subfolder as the hornsound config.txt.

horn.wav

idle.wav

**KIND: INTERIOR**

File location - world\custom\Interiors\

This is the traincar interior.

Referenced by the *interior* tag in a traincar config.txt

Green text = New to UTC

Config.txt:

```
kuid <KUID:####:#####>
kind interior
mesh interiormesh.pm (main .pm file)
camera 1.254, -0.053, 0.713
cameralist
{
camera0 -0.779, 0.035, 0.665, -1.16, -0.142
camera1 0.688, 0.1, 0.666, 0.057, -0.006
camera2 1.254, -0.053, 0.713
camera3 -1.169, -0.111, 0.725, 0.83, -6.413
camera4 -0.56, -0.031, 0.6, -13.732, -6.389
}
cameradefault 2
attachment {

    (Refer to additional interior config
    files provided in .zip file as
    different loco types have different
    requirements)

}
obsolete-table {
}
username
description ""
trainz-build 1.5
category-class AC
category-region-0 AT
category-era-0 1980s
```

**Breakdown:****mesh**

.pm model file

**camera**

SP3 camera position relative to a.cabfront (0,0,0 = left/right, front/back, up/down)

**cameralist**

Multiple in-cab camera positions relative to a.cabfront. This section ignored by SP3.

0,0,0,0,0 =left/right, front/back, up/down, yaw, pitch

To determine these variables add *-freeintcam* to the *trainzoptions.txt*. Pan around the interior using arrow keys and mouse. Co-ordinates at bot./left.

**cameradefault**

The in-cab camera view Trainz defaults to when entering the cab. (Generally should be the same as that of *camera*). This section ignored by SP3.

**attachment**

Start of attachments section. Specifies additional meshes and types inserted at specified attachment points within the main .pm model.

**INTERIOR ATTACHMENT TYPES:****pantograph\_lever**

Pantograph lever/switch. For raising and lowering pantographs on electric locos.

**horn**

Locomotive's Horn

**independantbrake\_lever**

Independent (Loco) brake lever

**reverser\_lever**

Reverser lever (Forward/Neutral/Reverse)

**throttle\_lever**

Throttle / power handle

**trainbrake\_lever**

Train brake lever - self lapping

**trainbrakelap\_lever**

Train brake lever with lap position.

**dynamicbrake\_lever**

For selecting dynamic brake

**bplocomain\_needle**

Main reservoir pressure needle

**bploco\_equalizer**

Equalising reservoir pressure needle

**bptrainbrakepipe\_needle**

Brake pipe pressure needle

**bptrainbrakecylinder\_needle**

Brake cylinder pressure needle

**speedo\_needle**

Speedometer needle

**ampmeter\_needle**

Power meter needle

**flow\_needle**

Flow gauge needle

**windows**

Textured mesh with low opacity (semi-transparent) to give impression of reflection. This mesh has the same 3D origin point as the main .pm model, therefore does not require an attachment point

**wheelslip\_light**

A warning light mesh that is only visible when the locomotive loses traction. This mesh has the same 3D origin point as the main .pm model, therefore does not require an attachment point

**switch0, switch1 etc**

Switches

**light\_switch**

Headlight switch

You can also attach miscellaneous meshes to attachment points. They have no current function in Trainz but they look pretty groovy ☺

E.g. a swivel chair

```
swivel_chair {
    kind lever
    mesh chair.pm
    att a.chair1
    limits 0, 8
    angles 6.8, -6.8
}
```

**INTERIOR ATTACHMENT VARIABLES:**

Kinds: **lever**

Lever's, switches, dials etc

**needle**

Guage needles (I.e. Speedo, brake pres.)

**pullrope**

Pullrope horn as in the F7

**light**

Wheelslip light

**mesh**

.pm file inserted

**att**

Attachment point where mesh is inserted. If no attachment point is specified the mesh will be inserted at a.cabfront (the same insertion point as main mesh)

**limits**

Mathematical boundaries Trainz uses determine the objects function. These values vary as different objects use different mathematical units. Generally use the default values used in the config files provided.

**angles**

Rotational boundaries in *radians* relative to its attachment point.

**notches**

The position of notches within the angle boundaries. These are represented as decimal points between and including 0 and 1.

**notchheight**

The size of the notches specified.

**radius**

The notch position relative to the attachment point.

**mousespeed**

This controls the use of the mouse on screen. Use this to adjust the push/pull functioning of levers and dials for example.

**opacity**

Used for the window mesh to give transparency (and the impression of reflection).

**KIND: PANTOGRAPH**

File location - \World\custom\pant\

These are the animated mechanisms on the roof of electric locomotives that conduct to an electric catenary (wires) above.

Referenced by the *pantograph* tag in a traincar config.txt

Config.txt:

```
kuid <KUID:###:#####>
kind pantograph
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class AL
category-region-0 US
category-era-0 1960s
```

**KIND: WATER**

File location - world\custom\environment\

Config.txt

```
kuid <KUID:###:#####>
kind water
region Britain
normal Water1
reflection Water1_R
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class EW
category-region-0 UK
category-era-0
```

**Breakdown for Water Config.txt:****region**

surveyor region.

**normal**

name of image for water texture, file should be 128 x 128 x 24bit tga.

**reflection**

name of image file for reflection, file should be 128 x 128 pixel x 256 color bitmap.

**KIND: ENVIRONMENT**

File location - world\custom\environment\

Config.txt:

```
kuid <KUID:###:#####>
kind environment
region Britain
normal mediumclouds
storm mediumClouds_Storm
night mediumClouds_Night
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class ES
category-region-0 UK
category-era-0
```

**Breakdown for Environment Config.txt:**

*region* – surveyor region.

*normal* – name of image file for normal sky, file should be 256 x 256 pixel 24bit tga file.

*storm* – name of image file for stormy sky, file should be 256 x 256 pixel 24bit tga file.

*night* – name of image file for night sky, file should be 256 x 256 pixel 24bit tga file.

**KIND: MAP**

File location - world\custom\maps\

The config.txt for maps are automatically generated by Trainz Surveyor.

You can add a soundscript to the config if desired such as the example below. Refer Soundscripts

Config.txt :

```
kind map
kuid <KUID:###:#####>
username Britain
workingscale 0
workingunits 0
water <KUID:-1:8009>
region Britain
soundscript {
    morning {
        ambient 1
        value-range 1, 0.1
        volume 0.3
        sound {
            ctry_day_1.wav
        }
    }
    night {
        ambient 1
        value-range 0, 0.9
        volume 0.3
        sound {
            night_loop.wav
        }
    }
}
```

**KIND: GROUNDTEXTURE**

File location - world\custom\ground\

Config.txt :

```
kuid <KUID:###:#####>
kind groundtexture
region Britain
rgb 112, 115, 59
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class GL
category-region-0 UK
category-era-0
```

**Breakdown for Groundtexture Config.txt:****region**

surveyor region.

**rgb**

color to be used in minimaps

**KIND: SCENERY**

File location - World\custom\scenery\

Config.txt :

```
kuid <KUID:###:#####>
kind scenery
region Britain
type Foliage
light 1
nightmode home
night Custom_Object_Nightwindows
autoanimation 1
kuid-table {
    0 <KUID:###:#####>
    1 <KUID:###:#####>
}
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class FS
category-region-0 UK
category-era-0
```

**region**

Surveyor region.

**type**

Surveyor type.

**light**

Sets lighting to be used for object to be ambient or directional. 0 sets ambient lighting and object is light by general light value, 1 sets directional light which is affected by the position of the sun.

**nightmode**

Optional *home*, *lamp* or *constant*. *Home* switches on night effect at dusk and off sometime during the night. *Lamp* switches the night effect on from dusk to dawn. *Constant* lights are on day and night.

**night**

name of object to be shown for night effect, stored in subfolder. Refer Chapter 4, SCENERY OBJECTS WITH LIGHTS AT NIGHT

**autoanimation**

This tag not required if no animation is present.

**1** = contains animation



**KIND: TRACK – RAILS**

File location - World\custom\track\

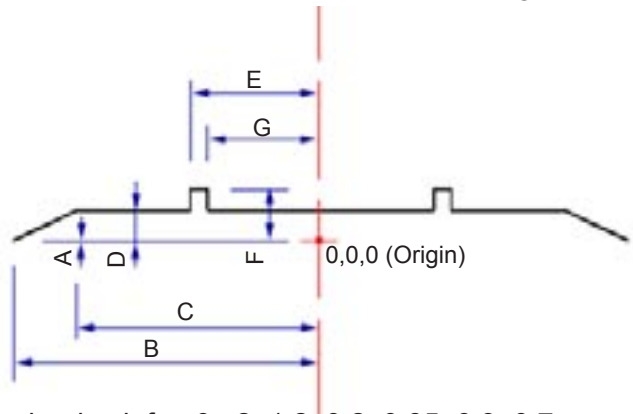
This is used for creating rails.

Config.txt :

```
kuid <KUID:###:#####>
kind track
region Britain
type Rails
rgb 255,200,0
length 4
istrack 1
width 4
chunky_mesh mstand_tex
chunky_info 0, 2, 1.2, 0.2, 0.85, 0.3, 0.7
kuid-table {
    0 <KUID:###:#####>
    1 <KUID:###:#####>
}
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class TR
category-region-0 UK
category-era-0
```

**chunky\_info**

These values (in metres) define the shape of the mesh created for the track. See drawing below:



```
chunky_info 0, 2, 1.2, 0.2, 0.85, 0.3, 0.7
chunky_info A, B, C, D, E, F, G
```

**type**

Surveyor type.

**region**

Surveyor region.

**rgb**

Color used for display in mini map

**length**

Length of track piece

**istrack**

Sets whether the track is a rail for trains or not.

**1** = This is a rail track

**width**

Width of track in meters

**chunky\_mesh**

Name of texture to apply to rail

**KIND: TRACK – ROAD**

File location - world\custom\track\

This is used for creating roads.

config.txt:

```
kuid <KUID:###:####>
kind track
region Australia
length 5
grounded 0.4
istrack 0
width 7.9
bendy 1
isroad 1
carrate 55
uncached_alphas 1
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class SR
category-region-0 AU
category-era-0
```

**uncached\_alphas**

This is used in certain situations to improve alpha sorting. This should only be set to 1 for tracks that use an alpha texture and are always placed flat near the ground (and are not used on bridges or turntables).

**Breakdown:****length**

Length of track segment in meters

**grounded**

Height in meters for the road to be offset from terrain

**istrack**

**0** = This is not rail tracks

**width**

Width of track mesh in meters.

**bendy**

Switches how track is bent on corners, set as 1 allows the mesh to be deformed as the spline is bend around corners.

**isroad**

Specifies track is a road with cars, set to 1 for cars to appear on road.

**carrate**

Defines traffic density on road (minimum seconds between each car generated).

**0** = No traffic.

Number must be greater than 3.

**KIND: BRIDGE – BRIDGE**

File location - world\custom\track\ or  
world\custom\splines\

This kind is used for creating road and rail bridge.

Config.txt :

```
kuid <KUID:###:#####>
kind bridge
type Bridges
region Britain
length 20
bridgetrack <KUID:-1:100395>
trackoffsets -2.5,2.5
height -8
rgb 200,100,0
casts_shadows 1
istrack 1
initiator dark_stone_arch_2t_start
terminator dark_stone_arch_2t_end
endlength 40
kuid-table {
    0 <KUID:###:#####>
    1 <KUID:###:#####>
}
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class TB
category-region-0 UK
category-era-0
```

**Breakdown:****type**

Surveyor type – eg. bridge, tunnel or rail.

**region**

Surveyor region.

**length**

Length in meters of each bridge piece

**bridgetrack**

Kuid for the type of rail or road used on bridge.

**trackoffsets**

Distance in meters the rail/s are attached to the center of the bridge spline. Any number of tracks can be attached to the spline, only splines with the same track offsets can be connected together.

**height**

Height from the track level to the base of the bridge supports, should be negative for bridges.

**rgb**

Color used for display in mini map

**casts\_shadows**

Defines whether or not the shadows are cast.

**0** = shadows off

**1** = shadows on

If shadows are on there needs to be a *bridge\_shadow.im* model in a subfolder for the bridge and the initiator and terminator segments (if they are used).

**istrack**

**0** = This is a road bridge

**1** = This is a rail bridge

**Initiator**

Name of model to use at start of bridge, placed in subfolder with same name.

**terminator**

Name of model to use at end of bridge, placed in subfolder with same name.

**endlength**

Length in meters of the initiator and terminator models.

**KIND: BRIDGE – TUNNEL**

File location - world\custom\track\ or  
world\custom\splines\

This kind is used for creating road and rail tunnels.

Config.txt :

```
kuid <KUID:###:#####>
kind bridge
type Tunnels
region Australia
length 20
bridgetrack <KUID:###:#####>
trackoffsets -4.5, 4.5
height 8
rgb 180, 180, 180
istrack 1
initiator oz_tunnel_start
terminator oz_tunnel_end
endlength 20
kuid-table {
    0 <KUID:###:#####>
    1 <KUID:###:#####>
}
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class TB
category-region-0 AU
category-era-0
```

**rgb**

Color used for display in mini map

**istrack**

**0** = This is a road bridge

**1** = This is a rail bridge

**Initiator**

Name of model to use at start of bridge, placed in subfolder with same name.

**terminator**

Name of model to use at end of bridge, placed in subfolder with same name.

**endlength**

Length in meters of the initiator and terminator models.

**Breakdown:****type**

Surveyor type – eg. bridge, tunnel or rail.

**region**

Surveyor region.

**length**

Length in meters of each bridge piece

**bridgetrack**

Kuid for the type of rail or road used on bridge.

**trackoffsets**

Distance in meters the rail/s are attached to the center of the bridge spline. Any number of tracks can be attached to the spline, only splines with the same track offsets can be connected together.

**height**

The height value for tunnels should be positive and greater than the height of the ceiling of the tunnel, but less than the height of the tunnel entrance structure.

**KIND: BRIDGE – DOUBLE TRACK**

File location - world\custom\track\

This kind can also be configured to create splines that can be used for placing two or more tracks using the *trackoffsets* tag.

Config.txt :

```
kuid <KUID:###:#####>
kind bridge
type Rails
region Australia
length 20
bridgetrack <KUID:-1:100396>
trackoffsets -2.5,2.5
height 0
rgb 255,200,0
istrack 1
kuid-table {
    0 <KUID:###:#####>
    1 <KUID:###:#####>
}
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class TB
category-region-0 UK
category-era-0
```

**istrack**

**0** = This is a road bridge

**1** = This is a rail bridge

**Breakdown:****type**

Surveyor type – eg. bridge, tunnel or rail.

**region**

Surveyor region.

**length**

Length in meters of each bridge piece

**bridgetrack**

Kuid for the type of rail used on bridge.

**trackoffsets**

Distance in meters the rail/s are attached to the center of the bridge spline. Any number of tracks can be attached to the spline, only splines with the same track offsets can be connected together.

**height**

**0** is used for double tracks.

**rgb**

Color used for display in mini map

**KIND: MOSPEEDBOARD**

File location - World\custom\trackside\

This is a speed limit sign.

Config.txt:

```
kuid <KUID:###:#####>
kind mospeedboard
trackside -2.5
speedlimit 5.56
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class WS
category-region-0 UK
category-era-0
```

**Breakdown:****trackside**

This is a value that is the distance in meters the object is placed relative to the center of the track. Negative values will put the object on the left side of the track, and positive values will appear on the right.

**Speedlimit**

This value is the maximum speed allowed in meters per seconds.

To convert miles per hour to meters per second multiply by a conversion factor of 0.447.

For example 10mph is 4.47 m/s.

To convert kilometers per hour to meters per second multiply by a conversion factor of 0.278.

For example 10kph is 2.78m/s.

**KIND: MOSIGNAL**

File location - world\custom\trackside\

Refer to config files attached within ZIP file.

Config.txt

```
kuid <KUID:####:#####>
kind mosignal
light 1
trackside -2.7
function TrackSignal
region Britain
name "02"
fontsize 0.07
fontcolor 255,255,255
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class WA
category-region-0 UK
category-era-0
```

**Breakdown:****region**

Surveyor region.

**light**

Sets lighting to be used for object to be ambient or directional. 0 sets ambient lighting and object is light by general light value, 1 sets directional light which is affected by the position of the sun.

**trackside**

Origin offset from track center expressed in metres.

**function**

Must be set to TrackSignal

**name**

Default number board entry

**fontsize** 0.07

Font size for number boards

**fontcolor** 255,255,255

R.G.B. font colour for number boards

**SIGNALS**

The next section of the config explains which aspects the signal is capable of displaying, and also which light points are activated when each state is displayed. The supported aspects are configured by reference number as follows...

0	STOP
1	STOP THEN PROCEED
2	CAUTION AND LEFT DIVERGE
3	CAUTION AND RIGHT DIVERGE
4	CAUTION
5	PROCEED AND LEFT DIVERGE
6	PROCEED AND RIGHT DIVERGE
7	ADVANCED CAUTION
8	PROCEED

The following two aspects are only used for scenarios....

9	SLOW
10	MEDIUM SPEED

The aspect section of the config.txt is arranged as follows.....

```
signals
{
0 {
    light 7
}
2 {
    light 6,0,1,2,3,4
}
4 {
    light 6
}
}
```

*(dont forget the 'space' between the number and bracket)*

Looking at the example above, under the heading 'signals' we see the states the signal is capable of displaying in the left column. From this extract we can see that this signal is capable of displaying aspects 0, 2 & 4.

When aspect 0 (stop) is displayed, light point 7 is activated.

When aspect 2 (caution left) is displayed, light points 6,0,1,2,3,4 are activated

When aspect 4 (caution) is displayed, light point 6 is activated.

## LIGHTS

Each light point needs to have a corona associated with it. Coronas are stored in each signal object's directory alongside its textures. Examples have been packaged within the zip file this document was located.

```
lights
{
0 {
corona corona_white.tga
}
1 {
corona corona_white.tga
}
2 {
corona corona_white.tga
}
3 {
corona corona_white.tga
}
4 {
corona corona_white.tga
}
5 {
corona corona_green.tga
}
6 {
corona corona_yellow.tga
}
7 {
corona corona_red.tga
}
8 {
corona corona_white.tga
}
9 {
corona corona_white.tga
}
}
```

*(dont forget the 'space' between the number and bracket)*

Looking at the example above, under the heading 'lights' we see the light points that are attached to the 3D model. This model has 10 of them, they are named a.light0 to a.light9.

*From the signals section we know that when aspect 0 (stop) is displayed, light point 7 is activated.*

Looking at the extract left...

When light point 7 is activated, it displays corona red.

*When aspect 2 (caution left) is displayed, light points 6,0,1,2,3,4 are activated*

When light point 6 is activated, it displays corona\_yellow.

When light points 0 – 4 are activated, each displays corona\_white.

Signal placement is very important for correct operation of the system. There are some rules to consider while signalling your map which if not observed may cause problems with getting the correct aspects to display.

There are also various departures from prototypical operation which should be considered when designing new signalling, and also when installing it into a map.

Please refer to the SP3 Signalling document for further details. Default location: \Trainz\Docs



**KIND: MOJUNCTION**

File location - World\custom\trackside\

This is used for creating junction control levers.

Config.txt :

```
kuid <KUID:###:#####>
kind mojunction
region Australia
trackside 2
light 1
mode0 lever1
mode1 lever2
soundscript
{
    toggle
    {
        trigger toggle
        distance 5, 100
        nostartdelay 1
        repeat-delay 1
        sound
        {
            points.wav
        }
    }
}
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class WX
category-region-0 AU
category-era-0
```

**Breakdown:****region**

Surveyor region.

**trackside**

This is a value that is the distance in meters the object is placed relative to the center of the track. Negative values will put the object on the left side of the track, and positive values will appear on the right.

**light**

Sets lighting to be used for object to be ambient or directional. 0 sets ambient lighting and object is light by general light value, 1 sets directional light which is affected by the position of the sun.

**mode0**

The model name, located in subfolder, of the initial junction. Example refers to a file lever1\lever1.im

**mode1**

The model name, located in subfolder, of the switched junction. Example refers to a file lever2\lever2.im

**soundscript**

Soundscripts for mojunction objects can be activated with toggle trigger message as in example. Refer to Soundscripts section (page 46)

**KIND: MOTURNTABLE**

File location - world\custom\scenery\

This is a turntable object.

Config.txt :

```
kuid <KUID:###:#####>
kind turntable
region Australia
type Trackside
light 1
mode0 oz_turntable2
mode1 oz_turntable2_spinner
angle 0,165,180,345
track <KUID:-1:100966>
snapmode 2
dighole 3,3
kuid-table {
    0 <KUID:###:#####>
    1 <KUID:###:#####>
}
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class TR
category-region-0 AU
category-era-0
```

**Breakdown:****type**

Surveyor type.

**region**

Surveyor region.

**light**

Sets lighting to be used for object to be ambient or directional. 0 sets ambient lighting and object is light by general light value, 1 sets directional light which is affected by the position of the sun.

**mode0**

The name of the main turntable object, model located in subfolder. Example refers to oz\_turntable2\ oz\_turntable2.im

**mode1**

Name of the rotating turntable part, model located in subfolder. Example refers to oz\_turntable2\_spinner \ oz\_turntable2\_spinner.im

**angle**

Specifies the angles at which the turntable stops.

**track**

Kuid for track to be attached to turntable

**snapmode**

Specifies the alignment of the turntable to the surveyor grid. 1 = origin snaps to grid (use for removing even dighole values), 2 = origin snaps to the center of a grid square (use for odd dighole values)

**dighole**

Specifies the number of grid segments (length, width) to be removed from the surveyor grid to accommodate the turntable pit.

**KIND: MOCROSSING**

File location - world\custom\scenery\

This is a level crossing.

Config.txt :

```
kuid <KUID:###:#####>
kind mocrossing
region Australia
type Trackside
track <KUID:-1:100396>
road <KUID:-1:100409>
mode0 level_crossing_1track
kuid-table {
    0 <KUID:###:#####>
    1 <KUID:###:#####>
}
obsolete-table {
}
username
description " "
trainz-build 1.5
category-class TR
category-region-0 AU
category-era-0
```

**Breakdown:****type**

Surveyor type.

**region**

Surveyor region.

**track**

Kuid for track to be used on crossing.

**road**

Kuid for road to be used on crossing.

**mode0**

Name of the model for the crossing object, stored in subfolder of same name. Example refers to level\_crossing\_1track\ level\_crossing\_1track.im

**KIND: ACTIVITY**

File location - world\custom\scenarios\

Config.txt :

```

kind activity
kuid <KUID:-14:160>
username Highland Valley (DCC)

scriptlibrary SP3S1DCC
scriptclass SP3S1DCC

driver-settings
{
  autopilotmode 0
  startingtime 0.4
  timerate 1
  deraillevel 0
  showhelp 0
  controlmethod 0
  weather 3
  changeability 1
}
kuid-table
{
  highland_valley      <KUID:-12:132>

  f7_sfred              <KUID:-1:1>
  atsf_chair            <KUID:-1:100160>
  atsf_pullman_pine    <KUID:-1:100163>
  atsf_baggage         <KUID:-1:100159>
  cflow_fert           <KUID:-1:100012>
  prr_fm_tuscan        <KUID:-1:100017>
  40ft_boxcar          <KUID:-1:100085>
  pdhc_babyruth        <KUID:-1:100066>
  4bhopper_il          <KUID:-1:100929>
  50ft_boxcar          <KUID:-1:100086>
  gatx_pennsalt        <KUID:-1:100092>
  60ft_boxcar          <KUID:-1:100087>
  sd40_2_santafe       <KUID:-1:100871>
  4bhopper_il_coal_full <KUID:-1:101224>
  foundry_car          <KUID:-1:101220>
}
description "Take contol of the morning
passenger service to Highland Valley
stopping at all stations and return to
Greenwood. Bad weather is forecast so
drive with care.

Service : Highland Valley Passenger
Train No. : 7528
Consist : F7A + 5 cars
Weight in Tow : 300t
Total Length : 490"
```

**Breakdown:****username**

Name of scenario displayed in trainz

**scriptlibrary**

The name of the .gsl (compiled script) library on disk, without the ".gsl" extension.

**scriptclass**

The name of the scenario class within the script file.

**driver-settings{}**

Specify the settings of this scenario, similar to Driver's 'settings' screen. Allows you to set the weather, control method (0 – dcc, 1 – cabin controlled) etc.

autopilotmode	0=of 1=on
startingtime	0..1 (0.5=midday)
timerate	1=real-time
deraillevel	0=none 1=arcadE 2=realistic
showhelp	0=ofF 1=on
controlmethod	0=dcc 1=cabin
weather	0=clear 1=cloudy 2=drizzle 3=rain 4=stormy 5=light snow 6=medium snow 7=heavy snow
changeability	0=none 1=periodic 2=extreme

**kuid-table{}**

A list of named assets used in the scenario. Scripts refer to assets (eg trains) by the names in this table.