

Resource Types

The game resources as well as the mod resources consist of code files (scripts, configurations, ...), 3D files, texture files and soundfiles. This page describes the structure of the most common ones.

Code Files

A lot of information is stored in code files. Code files in Transport Fever 2 usually use the following file extensions:

- .lua for scripts
- .lua for configurations
- .con for constructions
- .vs / .fs for [shaders](#)



All text files (models, meshes, scripts etc.) should be encoded in **UTF-8**. Also, it is recommended to use lower-case file names to avoid problems on case-sensitive OSes / file systems.

.lua for scripts

Script files usually return a struct with several functions that can be imported and used from elsewhere:

```
local scriptstruct = { }

function scriptstruct.functionname1(parameter1, ...)
    -- function code
end
...

return scriptstruct
```

Most of these files can be found in the `res/scripts/` directory. See the documentation of [script modding](#) for further details.

.lua for configurations

Configurational files usually consist at least of the `data ()` function that can be retrieved from elsewhere:

```
function data()
return {
    -- content of data struct depending on type of configuration
}
end
```

Most of these files can be found in the `res/config/` directory.

.con for constructions

Constructions use the file ending `.con` but basically their internal structure is similar to configurational files:

```
function data()
return {
    type = ..., -- a string constant used to distinguish between the different
types of constructions
    description = { -- name and description for the menu
        name = ...,
        description = ...
    },
    buildMode = ..., -- used to select the build mode
    categories = { ... }, -- used to filter elements in submenus
    order = ..., -- used to order the elements in the menu
    skipCollision = ..., -- skip collision (useful for decorative assets)
    autoRemovable = ..., -- can this construction be removed when other
elements should be built at this position
    updateFn = function(params)
        local result = { }

        ... -- fill the result with content

        return result
    end
}
end
```

Most of these files can be found in the `res/construction/` directory. See the documentation of [constructions](#) for further details.

3D Files

The model resources consist of the following file types:

- `.mdl` for model definitions
- `.msh` for mesh index files
- `.msh.blob` for mesh 3D data

- .mtl for material definitions
- .ani for extracted animations

.mdl for model definitions

Model files are a special kind of configuration file as well, thus their structure is similar with a `data()`-function.

```
function data()
return {
    boundingInfo = { ... }, -- optional bounding box used e.g. for render
    borders
    collider = { ... },      -- optional collider for collision calculation
    lods = { ... },          -- geometric information for 3D data and textures
    metadata = { ... },      -- metadata depending on model type
    version = 1,             -- required to distinguish from the old Transport
    Fever 1 model format
}
end
```

Detailed information about bounding boxes, colliders and the lod tree can be found in the [model documentation](#).

.msh / .msh.blob for mesh data

The mesh files contain the three-dimensional mesh information known as the actual 3D model. A mesh can consist of more than one submesh. This data is stored in a binary blob file that contains all the positions, orientations etc. of every vertex, edge and face. To tell the engine where the right information is stored in the binary blob file, there is a second file with indices. This text based file can be edited with any text editor.

```
function data()
return {
    subMeshes = { ... },
    vertexAttr = { ... },
}
end
```

Both files share the same name, but the blob file has `.blob` as a suffix. For example, if the index file is called `cube_lod_0.msh`, then the blob file is called `cube_lod_0.msh.blob`. Read more about meshes and the internal structure of these files in the [mesh documentation](#).

.mtl for material definition

Materials contain the reference to texture files. The basic structure is a `data()` function as well.

```
function data()
```

```

return {
  order = 0, -- used to determine rendering order between materials of same
type
  params = { ... }, -- parameters vary depending on type
  type = "PHYSICAL_NRML_MAP",
}
end

```

There are many different material types available. For a full overview of available types, see the [material documentation](#).

.ani for extracted animations

If animations are extracted from .mdl files to external files, they are located in .ani files.

These files contain a `data()` function as most of the other resources:

```

function data()
return {
  times = { 0, ... }, -- timestamps in milliseconds from 0 as begin of
animation
  transfs = {
    { 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, }, -- transformation
matrix for each timestamp
    ...
  },
}
end

```

The number of timestamps must equal the number of matrices!

Texture Files

There are two common file formats used for textures in Transport Fever 2, uncompressed .tga files and compressed .dds files.



To ensure compatibility with ATI and Intel graphic cards, it is important that the texture resolutions use multiples of 2 for the sizes: 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096.

Keep in mind that larger textures occupy a lot more memory space. Therefore only use large textures when absolutely necessary.

TGA (TARGA)

.tga files usually are used for UI elements and smaller graphics. For Transport Fever 2 it is important to **not** use the RLE compression. ~~.tga files have to be uncompressed.~~ (This restriction is not valid anymore.) As their filesize is greater than the file size of a similar image with compressed .dds-format, it is recommended to not use .tga for larger images.

DDS (DirectDraw Surface)

The .dds file format superseded the .tga format for the model textures. In contrast to it, .dds textures for Transport Fever 2 are compressed and have a smaller filesize, reducing the memory usage.

The common compression types supported by Transport Fever 2 are:

- DXT1 with binary alpha information
- DXT3 with 4bit alpha
- DXT5 with interpolated alpha
- 3Dc for normal maps

While DXT1 has the smallest file size, it is only suitable where no transparency or full transparency (either 100% opaque or 100% transparent) is needed.

For normalmaps, the 3Dc ATI2A2XY standard is required. See [external tools documentation](#) for more information on tools supporting these compressions.

Sound Files

A detailed explanation on the requirements of sound files can be found in the [sound effects documentation](#).

Translations

A detailed explanation on translation and localization modding can be found in the [localizations documentation](#).

[Syntax](#)

[Mod Components](#)

From:

<https://www.transportfever2.com/wiki/> - **Transport Fever 2 Wiki**

Permanent link:

<https://www.transportfever2.com/wiki/doku.php?id=modding:resourcetypes>

Last update: **2020/10/21 14:17**

