

Creating vis files from d4 personalities

(This is NOT an official documentation but simply based on my personal experiences - it might be not complete or even simply wrong in some aspects. Please give feedback if you find anything what you think should be improved.)

Even though the old visualiser is now superseded by Capture it might be useful for some - and as personalities for the visualiser are not made any more, here are some hints how you can make them from d4 Titan personalities.

A pre-requisite is of course an up-to-date manual or at least DMX chart of the fixture you want to do files for.

1. Check if there really doesn't exist one

Use the search on <https://personalities.avolites.com/> to check if there really doesn't exist what you need - downloading is always faster than attempting to do yourself.

2. Update your personality library

You certainly do not want to work with old data. Thus, download and install the latest Titan Fixture Library from <https://personalities.avolites.com/>.

3. Prepare a skeleton folder structure

This process will produce a couple of files which all need to go to designated folders. It has proven to be useful to have such a folder structure while preparing vis personalities:

```
My Fixture
  // You may put the manual/DMX-Chart here //
Avolites
  Titan
    FixtureLibrary
  Visualiser
    Personalities
    Extensions
```

Example:

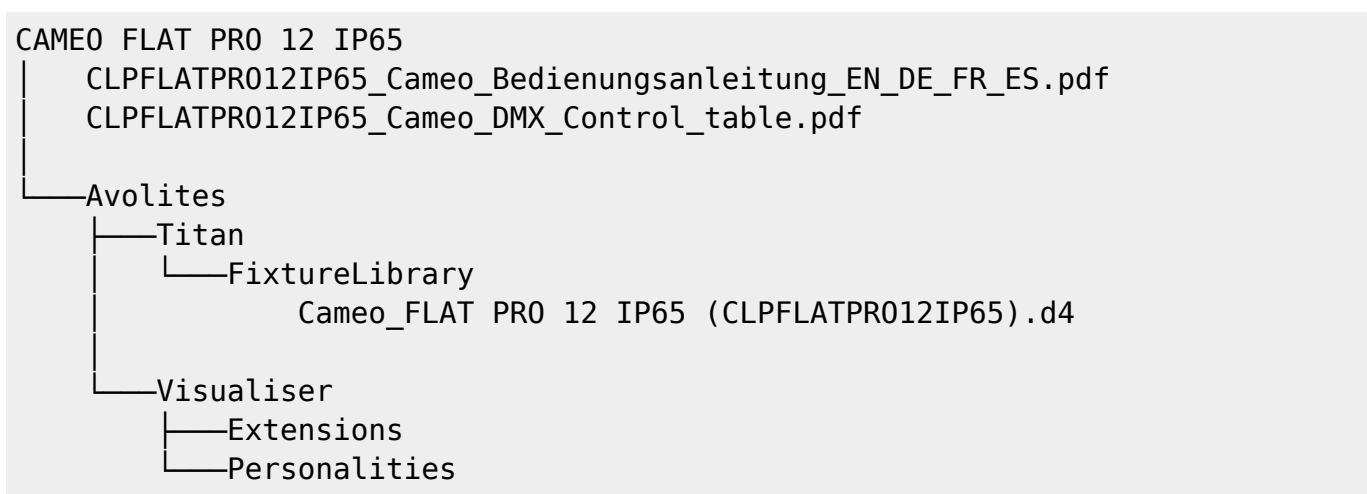


4. Identify the correct d4 file

We assume that a d4 personality for your fixture already exists. Luckily they usually have verbose file names - have a look in the personality folder (with the PC suite installed it is C:\Program Files (x86)\Avolites\Titan\FixtureLibrary). If you don't find what you need then search at <https://personalities.avolites.com/> or <https://www.avolites.de/downloads/d4find> for the correct filename. If you still don't find anything then request the personality for Titan or build it yourself (this might be explained on another day).

If you have found the correct d4 file copy it to \Avolites\Titan\FixtureLibrary in [the folder structure](#).

Now, our example looks like this:



Then doubleclick the d4 file to open it in the Titan Personality Builder.

5. Create/check Pearl references

In the personality builder expand the modes section (left panel). One by one click on each mode and check that there is a setting in the Classic Pearl Reference field bottom right. This makes the reference from Titan to the visualiser.

though 😎

From the File menu select Generate R20 Files...



The next prompt asks if you want to create Visualiser files as well. Confirm this with Yes.



Now, one by one, Personality Builder scans each mode. For each mode, two files are created: an .R20 file (like the classic pearl files) and a vis file which is the actual visualiser personality. However, things usually don't go that smoothly.

The next prompt asks where the file should be saved - usually we can just confirm this (the filename is created from our **unique** reference...):



Next, depending from the d4 file some prompts may be shown, e.g.

The channels assigned to wheels in this mode only utilise part of the DMX range.
Note: Visualiser will require complete range to work correctly.
Adjust limits to match?



Usually I select **No** here as we will check and correct the vis file manually anyway.

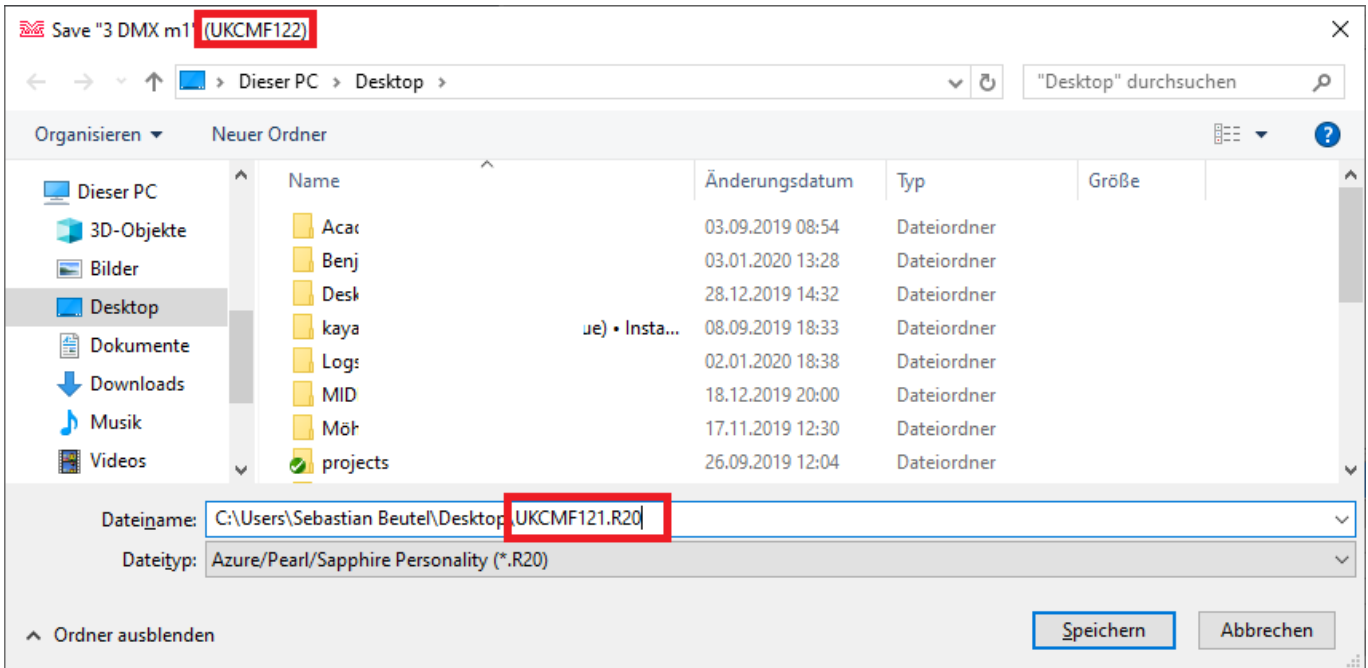
Another such prompt:

Fixture name is too long.
Please shorten it by 11 character(s):
.....



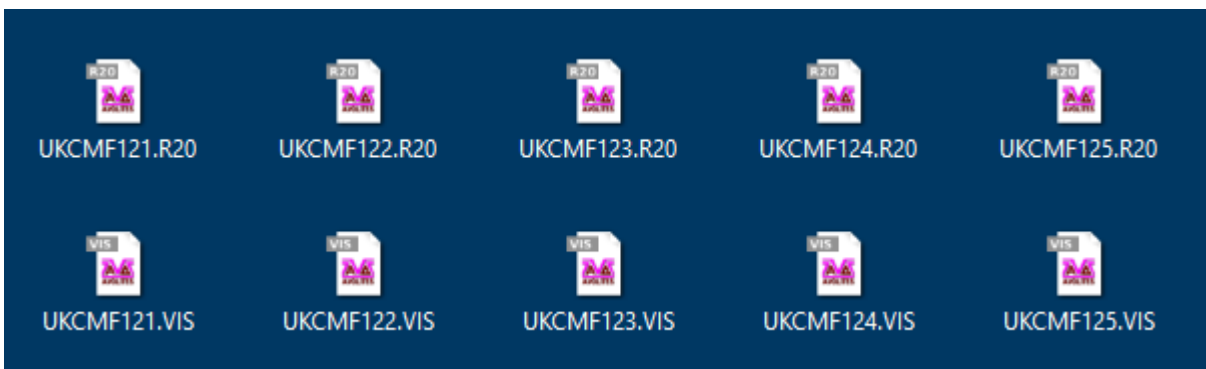
Here, you need to comply - shorten the given fixture name as only 11 characters can be displayed in the visualiser. This needs to be done per mode - it's a good idea to use something like 'm1', 'm2' or such to distinguish the modes. Here, I set the given name to FLAT PRO 12 m1.

Now the next mode is parsed and checked. Again we get the prompt asking for the filename. **This time it really needs to be checked and corrected** as it again suggest the name derived from mode 1. Change the filename for the one which refers to the reference of mode 2 (as suggested in the prompt's title bar):



Here it again suggests the filename UKCMF121 .R20 - however, the title bar tells that we have to change this to UKCMF122 .R20.

Proceed with this until all modes have been processed. Eventually, 2 files per mode will be created. Here, as all five modes have references and are exported, 10 files will be created:



Move all .R20 files to the folder Avolites\Visualiser\Personalities and all .VIS files to the folder Avolites\Visualiser\Extensions of your folder skeleton. In our example the result looks like this:

```
CAME0 FLAT PRO 12 IP65
| CLPFLATPR012IP65_Cameo_Bedienungsanleitung_EN_DE_FR_ES.pdf
| CLPFLATPR012IP65_Cameo_DMX_Control_table.pdf
```



7. Edit R20 files

Usually the R20 files - originally personalities for the Classic consoles - do not need to be edited. I had a couple of times where I needed to do this in the past in order to make the visualiser working properly. If I come across such a fixture again I will add this to this section.

Editing R20 files is done with a standard text editor. Make sure the editor really works with plain ASCII text without interfering with the character set, encoding or whatsoever. For starters, Windows Editor (a.k.a. Notepad) does work. Personally I prefer [Notepad++](#) - but there are many such programs available.

Most of the syntax is explained in the file itself. Only as a reference, here is a screenshot of such a file opened in Notepad++.

```

11 ; The device name, upto 11 characters, no spaces are allowed
12 DEVICE UKCMF121 mode=2 DMX 2 DMX Channels
13 NAME "Cameo" "FLAT PRO 12 ml"
14 ;
15 ;-----
16 ; The Type of instrument, 1 & M must required for backward compatibility
17 ; The middle number denotes the number of DMX channels used
18 TYPE 1 2 M
19 ;
20 ;-----
21 ; Mirror/Head resolution
22 MIRROR 0 0 0
23 ; | | |
24 ; | | | maximum tilt movement (in degrees)
25 ; | | | maximum pan movement (in degrees)
26 ; | | | 1 means moving mirror, 2 means moving head
27 ;
28 ;-----
29 ; Patching Information. (Upto 10 lines allowed)
30 ; Upto 10 lines of 159 chars are allowed including formulae
31 ; Variables used d=dmx number(1..512), b=handle(1..60), m=device name (11-char)
32 ; Allowed modifiers
33 ; +, -, *, /, (,) Basic arithmetic
34 ; % means remainder after divide
35 ; & means logical AND
36 ; ! means logical OR
37 ; R means reverse the bits of the current result (operates on a byte)
38 ; ^ means logical Exclusive OR (operates on a byte)
39 ; > means shift to the right
40 ; < shift to the left
41 ; # followed by number means get the value of that parameter
42 ; d DMX address
43 ; m Fixture name - 0 terminated character string

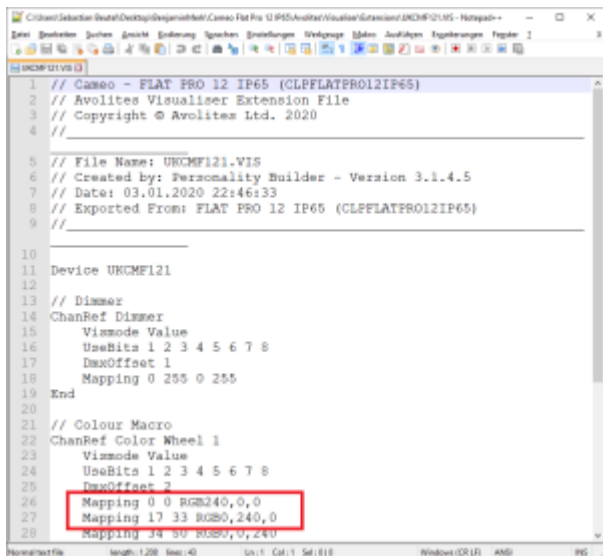
```

8. Edit vis files

Unlike the R20 files, the VIS files essentially always need to be edited before the fixture really works in the Visualiser.

Editing vis files is again done with a text editor (my recommendation: [Notepad++](#) as mentioned above). The settings are explained in chapter 10, 'Visualiser Definitions', of the [Visualiser Manual](#), from page 38 on. **This is a MUST READ.**

A vis file opened in an editor looks like this:



```
1 // Cameo - FLAT PRO 12 IP65 (CLPFLATPRO12IP65)
2 // Avolites Visualiser Extension File
3 // Copyright © Avolites Ltd. 2020
4 //
5 // File Name: UKCMF121.VIS
6 // Created by: Personality Builder - Version 3.1.4.5
7 // Date: 03.01.2020 22:46:33
8 // Exported From: FLAT PRO 12 IP65 (CLPFLATPRO12IP65)
9 //
10
11 Device UKCMF121
12
13 // Dimmer
14 ChanRef Dimmer
15 Vismode Value
16 UseBits 1 2 3 4 5 6 7 8
17 DmxOffset 1
18 Mapping 0 255 0 255
19 End
20
21 // Colour Macro
22 ChanRef Color Wheel 1
23 Vismode Value
24 UseBits 1 2 3 4 5 6 7 8
25 DmxOffset 2
26 Mapping 0 0 RGB240,0,0
27 Mapping 17 33 RGB0,240,0
28 Mapping 34 50 RGB0,0,240
```

I have already highlighted one thing to fix. But the basic structure is:

- there are many lines starting with a // double slash. These line are simple comments and have no function.
- there is a line like Device UKCMF121 - this is again the reference we started with
- there are some sections like this:

```
ChanRef Dimmer
Vismode Value
UseBits 1 2 3 4 5 6 7 8
DmxOffset 1
Mapping 0 255 0 255
End
```

- each section starts with the keyword ChanRef and the attribute name it is to control
- Vismode is the way how the attribute is being controlled (see the manual - apart from Value it can also be Speed and Strobe, usually in conjunction with a previously defined Value
- UseBits is the channel resolution. Nowadays as there are usually 8bit or 16bit channels it is either 1 2 3 4 5 6 7 8 for 8bit channels or 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 for 16bit channels
- DmxOffset is the dmx offset of this channel (you know what this is, right?)
- and finally there is one or more line starting with Mapping which define how the incoming value is treated/displayed in the Visualiser. The first two numbers are the incoming DMX values (here: from 0 to 255), the last two numbers are the outcome in the visualiser (here: from 0 to 255,

which, as it is the dimmer channel, is 0 to 100% dimmer level). **Again, please read the Visualiser Manual chapter 10 for details!**

In general I recommend to do as few editing as possible. The files are sensitive in many ways: an invalid character, symbol (point vs. comma!), or even new line (or missing new line), can yield unwanted results, from the fixture not showing in the visualiser at all to showing but acting up.

Here are some things which frequently need to be corrected:

8.1 complete dmx ranges

DMX ranges must be complete from 0 to 255 (8bit) or from 0 to 65535 (16bit) for the visualiser to work correctly.

In this example we find the colour wheel 1 definition like this:

```
ChanRef Color Wheel 1
  Vismode Value
  UseBits 1 2 3 4 5 6 7 8
  DmxOffset 2
  Mapping 0 0 RGB240,0,0
  Mapping 17 33 RGB0,240,0
  Mapping 34 50 RGB0,0,240
  Mapping 51 67 White
  Mapping 68 84 RGB240,100,0
  Mapping 85 101 RGB240,240,0
  Mapping 102 118 RGB0,240,240
  Mapping 119 135 RGB240,120,240
  Mapping 136 152 RGB240,140,140
  Mapping 153 169 RGB120,240,0
  Mapping 170 186 RGB240,0,240
  Mapping 187 203 RGB0,240,120
  Mapping 204 220 RGB240,120,0
  Mapping 221 237 RGB220,220,240
  Mapping 238 255 RGB240,240,200
End
```

There is a gap in the DMX values between the first range Mapping 0 0 RGB240,0,0 and the second range Mapping 17 33 RGB0,240,0. Solve this by changing the first range to Mapping 0 16 RGB240,0,0. Finish by checking all ranges that each one starts immediately after the previous one ends.

8.2 check for fundamental ranges (open/close/safe etc.)

Sometimes some attributes do not have the fundamental ranges e.g. to open or close the shutter, open gobo, stop rotation, etc.

Here, in the exported mode 2 visualiser file (UKCMF122.VIS) we find a definition like this:

```
// Shutter
ChanRef Dimmer
  Vismode Strobe
  UseBits 1 2 3 4 5 6 7 8
  DmxOffset 2
  Mapping 0 255 30 300
End
```

The line Mapping 0 255 30 300 defines that DMX values from 0 to 100% translate to a strobe frequency of 30 to 300 flashes per minute (0.5 ~ 5 Hz) - there is no 'Open' or 'No Strobe' range which is highly unlikely. A quick check in the manual reveals this:



3-CHANNEL MODE 1		
CHANNEL	VALUE	FUNCTION
CH1	000 - 255	Master dimmer (0 - 100%)
CH2	000 - 255	Stroboscope (rate 0 - 100%)
	000 - 080	Colour Macros
	000 - 004	Blackout

Obviously our visualiser file needs to be added the 0% strobe range. We add this range (**Strobe 1 is Shutter Open!**) and change the dmx start of the existing range so that the ranges read

```
Mapping 0 0 1 1
Mapping 1 255 30 300
```

Additionally we fix the colourwheel definition in the same way like we did for the first mode.

8.3 remove redundant ranges

9. copy to correct folders

10. test

From: <https://www.avosupport.de/wiki/> - AVOSUPPORT

Permanent link: https://www.avosupport.de/wiki/personality/create_vis_from_d4?rev=1578139873

Last update: 2020/01/04 12:11

