

CONFIGURATION GUIDE for CIRCULATION VJ software

GENERAL INFORMATION

In this guide you will find information how to prepare your media for your VJ set. Usually all parameters are SPACE separated. First you have to locate in folder in **\dat** appropriate subfolder to store media and edit **list.txt** configuration file(s). Each plugin got it's own format and logic how to structure your media.

FADERMAPS

Configuration file is **dat\circulation\fadermaps\list.txt**. This file includes a list of grayscale masks.

dat\circulation\fadermaps - folder with fadermaps
list.txt - fadermaps definitions

Every line in the configuration file describes one fadermap.

Configuration line consists from 2 compulsory parameters:

KEY FILE

- A..Z,0..9** shader's shortcut character
In case you don't want to assign shortcut character use "_" (underscopel) character.
- FILE** filename of fadermap image

example:

We want to add 3 grayscale images for mask transitions, gradient or time distort fx. Locate folder **\dat\circulation\fadermaps** and copy desired images:
mask1.tga
mask2.tga
mask3.tga

Now we define these images in **\dat\circulation\fadermaps\list.txt**

list.txt contains:

```
Q mask1.tga
W mask2.tga
_ mask3.tga
```

We defined 3 fadermaps. First two of them have assigned shortcut character, but last one got undefined shortcut character.

OFFSET MAPS

Configuration file is **dat\circulation\fadermaps\list.txt**. This file includes a list of grayscale masks used in PPX Offset filter.

dat\circulation\posteffects\offser - folder with maps
list.txt - offsets definitions

Every line in the configuration file describes one offset map.

Two images, ideally in render resolution, are needed per one offset map:

FILE1 FILE2

- FILE1** axis X offset map
- FILE2** axis Y offset map

example:

We want to add 2 offset masks for PPX Offset. Locate folder **\dat\circulation\posteffects\offset** and copy desired images:
off1_x.tga
off1_y.tga
off2_x.tga
off2_y.tga

Now we define these images in **\dat\circulation\posteffects\offset\list.txt**

list.txt contains:

```
off1_x.tga off1_y.tga
off2_x.tga off2_y.tga
```

This way we defined 2 offset maps. you will use them by enabling Offset filter in posteffects.



circulation

www.circulation.cc | www.satori.sk
© SATORI, s.r.o. 2002-2017

AVIPLAY

A configuration file is located in **dat\aviplay** folder. By default **dat\aviplay\list.txt** is searched and loaded. This file includes a list of videos assigned for use by AVIPLAY and you can setup default settings for each clip. You can set a different configuration list by using **setup** or in circulation.ini (tag:aviplay_list=<list_file>).

dat\aviplay - folder with videos and config
If not stated otherwise, via Circulation setup, your set is configured in the following file:

list.txt - videos config file
Edit circulation.ini tag **aviplay_list** to use a different list file.

Every line of the configuration file describes one video.
However, by the use of wildcard characted in filename, you can load more videos at once.

Compulsory part of the configuration are 3 parameters describing how are videos assigned to banks and key letters. Additional parameters may follow:

BANK KEY FILE [PARAM1] [PARAM2] [PARAM3] ...

- 0..99** number of BANK
- A..Z** KEY letter for the video
What is altogether 2600 combinations and available positions for your videos.
- FILE** followed by video's relative or absolute path.
In case of spaces inside file-path the whole string must be in "..."

Other parameters are voluntary and set various attributes.

```
loop video looping
pingpong ping pong looping
nooop don't loop video (default)
start_frame loop start frame
end_frame loop end frame
start_time loop start timecode
end_time loop end timecode
name= label your video (ID string)
group= assign video to group (ID string)
prev= previous video label for chaining (ID string)
next= next video label for chaining (ID string)
```

ID strings can NOT contain the following characters 'SPACE' and '_' otherwise the commands are parsed incorrectly.

```
deinterlace deinterlace video fields
odd_field deinterlace type
even_field deinterlace type
no_stretch don't scale video to render resolution
zoom_all no letterboxing of different aspect videos
```

mini_note= decimal midi note number to trigger video
Note: On event of given note plays video.
mini= decimal midi controller number to trigger video
Controller event must be 127 (FF).

example:

```
; the very basic and compulsory setup: <db> <ks> <file> <avi>
I Q hello.avi
; usage of absolute path and extra attributes
I W "c:\program files\101.avi" nooop deinterlace even_field
; looping 3 video in a row
I E vid3.avi name=one next=two start_frame=10 end_frame=30
I R vid2.avi name=two next=three
I T vid3.avi name=three next=one
```

If you wish to automatically assign key letters to videos there is simplified form. You define which bank you assign videos in with:

into_bank= number of bank 0..99
Now every line in config starting with "..." (2 times underscore) is assigned automatically to defined bank only.

example:

```
into_bank=3
- c1rc1.avi
- c1rc2.avi
- c1rc3.avi
; will be assigned 3 Q, 3 W, 3 E
```

If you don't care about destination bank and you want to fill consecutive banks with more videos, use

bank_free= number of bank 0..99 (default)
to declare assigning videos to first available free slot in defined bank and if full, follow to further banks.

You can add more files at once by the use of wildcard character. Other stated parameters will be used for all batch loaded videos.

example:

```
bank_free=10
- myvideo1*.avi pingpong group=myvids
; will load all 10 videos from subfolder to bank 10+ with pingpong loop and assigned to group "myvids".
```

You can use comments in config file:

```
; comment
# comment
/*
start of multiline comment
commenting ends here
*/
```

No comments possible end the end of valid configuration line. So this is not allowed:
I Q helloworld.avi // my first video

SOURCES & ENGINE MEDIA

PICBLUR

Main configuration file and images are located in **dat\picblur** folder.

By default **dat\picblur\list.txt** is searched and loaded. This file includes a list of images assigned for use by PICBLUR and other image based plugins. You can set a different subfolder where configuration list.txt will be loaded using **setup** or in circulation.ini (tag:picblur_dir=<directory>).

dat\picblur - folder with images
Define **picblur_dir** via setup to search and load images from subfolder localised in dat\picblur.

list.txt - images config

Single bank configuration.

Every line of main configuration file describes one image.

Compulsory part of the configuration are 2 parameters:

KEY IMAGE

- A..Z** KEY letter for the image
Use character "_" (underscopel) if you don't want to set key letter shortcut.
- IMAGE** followed by IMAGE's relative path.
No spaces allowed in filename or path.

Multiple banks configuration.

This way you may sort images to 20 available banks.

First, you have to create subfolders in **\dat\picblur** where you will locate sorted images per bank theme. In that subfolder create another list.txt describing KEY letter and IMAGE filename (separated by space).

Second, you have to edit main configuration and add banks definition with the following command:

bank, BANK, SUBFOLDER, [no load]

- bank** definition tag
- 0..19** BANK number
- PATH** relative path to folder with bank's images
- no load** don't preload images on start
no load is compulsory parameter!

If you want to add a lot of large images, you might run out of memory. Therefore load these images with **no load** tag. But in a such way, these images won't be available in other image based plugins.

Images configure in main configuration are automatically assigned to default bank 0. If no image are assigned to bank 0, then first available bank will be used as default.

example:

We want to have 3 banks of images: renders, photos and gifs. So we create 3 subfolders in **\dat\picblur**:

```
\dat\picblur\renders
\dat\picblur\photos
\dat\picblur\gifs
```

Then, we locate some images in these subfolders, e.g.:

```
\dat\picblur\renders\render1.png
\dat\picblur\renders\render2.png
\dat\picblur\renders\render3.png
\dat\picblur\photos\photo1.jpg
\dat\picblur\photos\photo2.jpg
\dat\picblur\photos\photo3.jpg
\dat\picblur\gifs\gif1.gif
\dat\picblur\gifs\gif2.gif
\dat\picblur\gifs\gif3.gif
```

In every subfolder we create a definition list.

```
\dat\picblur\renders\list.txt contains:
Q render1.png
W render2.png
E render3.png
```

```
\dat\picblur\photos\list.txt contains:
Q photo1.jpg
W photo2.jpg
E photo3.jpg
```

```
\dat\picblur\gifs\list.txt contains:
Q gif1.gif
W gif2.gif
E gif3.gif
```

Last step is to define main configuration list.

```
\dat\picblur\list.txt contains:
bank,1,renders
bank,2,photos,no load
bank,3,gifs
```

This way you have defined 3 banks of images. Photos bank was stated with "no load" parameters to save the memory.

You can use comments in config file:

```
; comment
# comment
No comments possible end the end of valid configuration line. So this is not allowed:
bank,3,gifs; gifs from net
```

SLIDER

Main configuration file is **dat\slider\list.txt**. This file includes a list of slideshows definitions.

dat\slider - folder with your slideshows
list.txt - slideshows definitions

Every line of main configuration file describes one slideshow.

Compulsory part of the configuration line are first 4 parameters:

KEY SUBFOLDER STILLTIME FADETIME [rand]

- A..Z,0..9** slideshow's KEY letter/number
- PATH** relative path to folder with slideshow
- NUMBER** stilltime for single image (in ms)
- NUMBER** crossfade time for images change (in ms)
- rand** enables randomizer order

rand is optional parameter!
No spaces allowed in filename or path.

Each slideshow is configured in it's subsequent subfolder.

Create a folder for each slideshow and store your desired images for it. Next step in this folder is to create **list.txt** file listing all the used images. Format is easy, just list your images' filenames one per line.

example:

We want to have 2 slideshows: trees and clouds.

So we create 2 subfolders in **\dat\slider**:
\dat\slider\trees
\dat\slider\clouds

We add some images there, e.g.:

```
\dat\slider\trees\stromy2.png
\dat\slider\trees\stromy2.png
\dat\slider\clouds\oblak1.jpg
\dat\slider\clouds\oblak2.jpg
\dat\slider\clouds\oblak3.jpg
```

In every subfolder we create a simple list of files:

```
\dat\slider\trees\list.txt contains:
stromy1.jpg
stromy2.jpg
stromy3.jpg
```

```
\dat\slider\clouds\list.txt contains:
```

```
oblak1.jpg
oblak2.jpg
oblak3.jpg
```

Last step is to define main configuration list.

```
\dat\slider\list.txt contains:
Q trees 2000 500
W clouds 5000 1000 rand
```

This way you have defined two slideshows. Trees slideshow is with 2 seconds for still, and half second for crossfade. Clouds slideshow has randomized order and 5s for still and 1s for crossfade.

You can use comments in config file:

```
; comment
# comment
No comments possible end the end of valid configuration line. So this is not allowed:
bank,3,gifs; gifs from net
```

SHADERS

Here you are about to configure fragment shaders for filtering.

Configuration file is **dat\circulation\shaders\list.txt**. This file includes a list of fragment shaders for GPU filtering of your rendering.

dat\circulation\shaders - folder with shaders
list.txt - shaders definitions

Every line in the configuration file describes one shader.

Compulsory part of the configuration line are first 2 parameters:

KEY FILE [par1] [par2] [par3] [par4]

- A..Z,0..9** shader's shortcut character
 - FILE** filename of shader
 - NUMBER** default parameter 1
 - NUMBER** default parameter 2
 - NUMBER** default parameter 3
 - NUMBER** default parameter 4
- par1, par2, par3, par4** are optional parameter!
No spaces allowed in filename or path.

example:

We want to add 3 vertex shaders for filtering. So locate folder **\dat\circulation\shaders** and copy desired shaders:
radial_blur.gpu
fish_eye.gpu
waveinterp.gpu

Now we add definitions for these shaders in **\dat\circulation\shaders\list.txt**

list.txt contains:

```
Q radial_blur.gpu 128 0 0 0
W fish_eye.gpu 32 32 0 0
E waveinterp.gpu 100 0 0 0
```

We defined 3 shaders with setting default parameters.