

Furcadia FOX File Format

Author: Aleksi Asikainen (sanct@furcadia.com)
Last Update: 2008-11-05
Scope: Description of FOX file format

General File Hierarchy

HEADER

- SHAPEBLOCK (shape #0)
 - SHAPEHEADER
 - (SHAPEEXT)
 - FRAMEBLOCK (frame #0)
 - FRAMEHEADER
 - (FRAMEEXT)
 - IMAGEDATA
 - FRAMEBLOCK (frame #1)
 - ...
 - STEPBLOCK (step #0)
 - STEPBLOCK (step #1)
 - ...
 - SHAPEBLOCK (shape #1)
 - ...

PNG Compression

Data Structure

Data type	Size	Name	Description
<u>HEADER</u>	28	header	FOX File header
num_shapes *	?	shapes	Shape data
<u>SHAPEBLOCK</u>			
EOF	EOF	EOF	EOF

HEADER

Data type	Size	Name	Description
char	4	magic	Always "FSHX" (0x46 0x53 0x48 0x58)
int	4	version	FOX format version. Currently 1, 2, or 3.
int	4	num_shapes	Total number of shapes in the FOX file
int	4	generator	File generator signature 0 – FSH Editor 1 – Dream upload process Values 0 – 131,072 reserved for DEP.
int	4	encryption	FOX encryption scheme or 0 for none. If not zero, reading of the file should be ceased.
int	4	reserved	Reserved for future
int	4	reserved	Reserved for future

SHAPEBLOCK

Data type	Size	Name	Description
SHAPEHEADER	8	shape_header	Shape-specific header data
SHAPEEXT	?	shape_extended_data	Extended shape-specific header data
num_frames * FRAMEBLOCK	?	frames	Frame header and image data
num_steps * STEPBLOCK	?	steps	KitterSpeak animation data

SHAPEHEADER

Data type	Size	Name	Description
u_short	2	flags	Shape-specific bit flags, may be any combination of the following: 1 – walkable 2 – gettable 4 – sittable
short	2	shape_no	If not -1, index of the shape this shape replaces.
u_short	2	num_frames	Total number of frames in the shape animation
u_steps	2	num_steps	Total number of steps in the shape animation

SHAPEEXT

This block of data **does not exist and should not be read** if FOX format version is 1 or 2. The file reader should ignore reading the SHAPEEXT block, unless FOX format version is 3 or above. The size of SHAPEEXT block **will differ** between FOX format versions.

Data type	Size	Name	Description
u_short	2	ext_data_size	The total size of the extended data block, including ext_data_size field.
TBA	TBA	TBA	TBA

FRAMEBLOCK

Data type	Size	Name	Description
FRAMEHEADER	18	frame_header	Frame-specific header data
FRAMEEXT	?	frame_extended_data	Extended frame-specific header data
IMAGEDATA or PNG DATA	image_data_size	image_data	Frame image data

STEPBLOCK

Data type	Size	Name	Description
u_short	2	step_type	Type of the step descriptor. Can be one of the following values: 1 – FRAME 2 – DELAY 3 – LOOP 4 – JUMP 5 – POSX 6 – POSY 7 – FURREX 8 – FURREY 9 – DRAW_FRONT 10 – DRAW_BEHIND 11 – AUTO_FRAME_DELAY 12 – STOP 13 – CAMERA_STATE 14 – RAND_FRAME_DELAY 15 – RAND_AUTO_FRAME_DELAY 16 – SHAPE_FRAME 17 – OPACITY

			18 – SLIDE_POSX 19 – SLIDE_POSY 20 – SLIDE_FURREX 21 – SLIDE_FURREY 22 – SLIDE_OPACITY 23 – SHOW_BGFRAME 24 – SHOW_FGFRAME 25 – SHOW_BGSHAPE 26 – SHOW_FGSHAPE 27 – HIDE_BG 28 – HIDE_FG
short	2	arg1 (value)	
short	2	arg2 (counter_max)	

FRAMEHEADER

Data type	Size	Name	Description
u_short	2	frame_format	Frame format, one of the following values: All versions: 1 – FORMAT_8BIT Only version 2 and above: 2 – FORMAT_BGR 3 – FORMAT_BGRA 7 – FORMAT_BGRA_RECOL
u_short	2	frame_width	The width of the frame in pixels
u_short	2	frame_height	The height of the frame in pixels
short	2	pos_x	Shape relative position
short	2	pos_y	Shape relative position
short	2	furre_pos_x	Avatar relative position
short	2	furre_pos_y	Avatar relative position
u_int	4	image_data_size	Total size of the image data block in bytes

FRAMEEXT

This block of data **does not exist and should not be read** if FOX format version is 1 or 2. The file reader should ignore reading the FRAMEEXT block, unless FOX format version is 3 or above. The size of FRAMEEXT block **will differ** between FOX format versions.

Data type	Size	Name	Min Version	Description
u_short	2	ext_data_size	3	The total size of the extended data block, including

				ext_data_size field.
u_char	1	opacity	3	Initial alpha level for the frame
u_char	1	compression_type	3	Frame compression type 0 – None 1 – PNG Compression

IMAGEDATA

The contents of the IMAGEDATA block depend on the value of frame_format and compression_type. For details about PNG compressed data, please see section [PNG COMPRESSION](#). The following data description assumes that compression_type has been set zero.

frame_format = FORMAT_8BIT

Data type	Size	Name	Description
8BPIXEL	frame_width * frame_height bytes	image_data	Image data, one byte per pixel

frame_format = FORMAT_BGR

Data type	Size	Name	Description
24BPIXEL	frame_width * frame_height * 3 bytes	image_data	Image data, three bytes per pixel

frame_format = FORMAT_BGRA

Data type	Size	Name	Description
24BPIXEL	frame_width * frame_height * 3 bytes	image_data	Image data, three bytes per pixel
APIXEL	frame_width * frame_height	alpha_mask	Alpha mask, one byte per pixel

frame_format = FORMAT_BGRA_RECOL

Data type	Size	Name	Description
24BPIXEL	frame_width * frame_height * 3 bytes	image_data	Image data, three bytes per pixel
APIXEL	frame_width * frame_height	alpha_mask	Alpha mask, one byte per pixel
RPIXEL	frame_width * frame_height	recolor_mask	Recolor mask, one byte per pixel

8BPIXEL

Data type	Size	Name	Description
unsigned char	1	pixel	A single-byte reference to the Furcadia color palette. Value 0 is considered transparent.

24BPIXEL

Data type	Size	Name	Description
unsigned char	1	B	Value BGR(0,0,0) is considered transparent unless accompanied with an alpha mask.
unsigned char	1	G	
unsigned char	1	R	

APIXEL

Data type	Size	Name	Description
unsigned char	1	A	Defines the level of translucency for the pixel (255 – opaque ... 0 – transparent)

RPIXEL

Data type	Size	Name	Description
unsigned char	1	recolor_mask	Defines to which re-coloring group the pixel belongs or 0 for none.

PNG COMPRESSION

If a frame has been compressed, the image data will be in PNG format. The PNG data **must** follow the specific PNG color type and bit depth of the given frame format as well as have matching image width and height to what has been described in the [FRAMEHEADER](#) block.

If the frame contains an alpha mask or a recolor mask, the mask data will follow as **separate image in PNG format**. Therefore a [FORMAT_BGRA_RECOL](#) frame will have **three** PNG format images – one PNG_COLOR_TYPE_RGB image followed by two PNG_COLOR_TYPE_GRAY (alpha mask and recolor mask, respectively) images.

frame_format = FORMAT_8BIT

Data type	Size	Name	Description
PNG_8BIT	?	image_data	

frame_format = FORMAT_BGR

Data type	Size	Name	Description
PNG_RGB	?	image_data	

frame_format = FORMAT_BGRA

Data type	Size	Name	Description
PNG_RGB	?	image_data	
PNG_8BIT	?	alpha_mask	

frame_format = FORMAT_BGRA_RECOL

Data type	Size	Name	Description
PNG_RGB	?	image_data	
PNG_8BIT	?	alpha_mask	
PNG_8BIT	?	recolor_mask	

PNG_8BIT

8 bit image data is saved without palette.

PNG Property	Value
Color Type	PNG_COLOR_TYPE_GRAY
Bit Depth	8

PNG_RGB

PNG Property	Value
Color Type	PNG_COLOR_TYPE_RGB
Bit Depth	8

Dragon's Eye Productions, Inc. reserves the right to change any practices, protocols, or formats described in this document without prior notice.