# FlickView2

# Objects

There is an object corresponding to each component of a .flc file. There are separate header objects not mentioned for each of the following objects.

## FlickObject

FlickObject is the primary object that acts as a container for the frames that make up the flick. Operations that have to be applied globally to all frames are performed by this object. Frames are manufactured by the FlickObject which acts as a FlickFrame factory. Since the number of frames is known in advance and fixed, a single call to AllocateFrames() allocates all the frame objects.

The GetFrame() method can be used to return an individual frame as a bitmap object.

All the frames can be saved as a single bitmap in .BMP or .PNG file format. This is primarily for debug purposes.

## FlickFrame

FlickFrame object takes care of managing the palette, image buffer and flick chunk objects. The Frame object is a container of chunks. The FlickFrame object allocates an image buffer that is a multiple of four bytes wide. This is needed to allow the image buffer to be passed directly to a .NET bitmap constructor.

In theory each frame could have it’s own palette. In practice a number of frames will share the same palette. There is an indicator in the frame object for when a new palette is needed for subsequent frames. Otherwise later frames use palette of the previous frame.

## FlickChunk

FlickChunk object handles processing different chunk types that compose the flick object. Only four chunk types are currently supported. COLOR\_256, COLOR\_64, BYTE\_RUN, and DELTA\_FLI. The FlickChunk handlers decompress the flick object into an image buffer.

# Serialization

## load()

The objects are serialized from the file system via load() methods. The load() method takes an input file stream reference as a parameter. There is currently no store() methods.